

**PENERAPAN ALGORITMA GENETIKA PADA
RELEVANCE FEEDBACK DALAM SISTEM TEMU
KEMBALI INFORMASI MENGGUNAKAN
*BINARY INDEPENDENCE MODEL (BIM)***

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Pada
Jurusan Teknik Informatika

Oleh:

BUDI SANTOSO

10751000115



**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU
PEKANBARU
2013**

**PENERAPAN ALGORITMA GENETIKA PADA
RELEVANCE FEEDBACK DALAM SISTEM TEMU
KEMBALI INFORMASI MENGGUNAKAN
*BINARY INDEPENDENCE MODEL (BIM)***

BUDI SANTOSO

10751000115

Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Sultan Syarif Kasim Riau

ABSTRAK

Kecepatan perubahan dan penambahan informasi menyebabkan dibutuhkannya suatu sistem yang dapat mengakses dan menyediakan informasi yang akurat dan relevan. Sistem temu kembali informasi merupakan sebuah sistem pencarian yang menemu kembalikan dokumen relevan sesuai dengan kata kunci yang diinputkan pengguna. Penerapan *relevance feedback* merupakan metode untuk meningkatkan hasil kemampuan dari kerelevanan dokumen yang dihasilkan oleh sistem temu kembali informasi. Pembangunan sistem temu kembali informasi dibangun dengan menerapkan *binary independence model* dan penerapan metode *relevance feedback* dengan menerapkan algoritma genetika. Koleksi dokumen yang dijadikan bahan penelitian adalah jurnal-jurnal teknologi dengan format *txt*. Pengujian kemampuan penerapan *relevance feedback* dalam sistem temu kembali informasi dilakukan dengan menggunakan *precision* dan *recall*. Hasil pengujian akan diketahui tingkat keakuratan dan kerelevanan dokumen hasil pencarian berdasarkan *query* pengguna. Rata-rata nilai *precision* sistem temu kembali informasi dengan *binary independence model* 61,53% dan rata-rata nilai *recall* 100%. Sedangkan nilai rata-rata *precision* penerapan *relevance feedback* dengan algoritma genetika 51.95% dan nilai rata-rata *recall* 100%, dimana hasil pengujian ini sangat dipengaruhi oleh jumlah dokumen, panjang *query* dan panjang dokumen yang dimasukan ke dalam sistem temu kembali informasi.

Kata kunci: Algoritma genetika, *binary independence model*, *relevance feedback*,, sistem temu kembali informasi.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR HAK ATAS KELAYAKAN INTELEKTUAL	iv
LEMBAR PERNYATAAN	v
LEMBAR PERSEMBAHAN	vi
ABSTRAK	vii
<i>ABSTRACT</i>	viii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xv
DAFTAR TABEL	xix
DAFTAR RUMUS	xxi
DAFTAR LAMPIRAN	xxii
DAFTAR SIMBOL	xxiii
DAFTAR ISTILAH	xxiv
BAB I PENDAHULUAN	I-1
1.1. Latar Belakang	I-1
1.2. Rumusan Masalah	I-3
1.3. Batasan Masalah	I-3
1.4. Tujuan	I-4

1.5. Sistematika Pembahasan	I-4
BAB II LANDASAN TEORI	II-1
2.1. Sistem Temu Kembali Informasi	II-1
2.1.1. Pengertian Sistem Temu Kembali Informasi	II-1
2.1.2. Istilah Sistem Temu Kembali Informasi	II-2
2.1.3. Pengukuran Kemampuan Dalam Sistem Temu Kembali Informasi	II-3
2.1.4. Arsitektur Sistem Temu Balik Informasi	II-4
2.1.5. Koleksi Dokumen (<i>Corpus</i>)	II-5
2.2. Model-Model Sistem Temu Kembali Informasi	II-9
2.2.1. Model Boolean	II-9
2.2.2. Model Ruang Vektor.....	II-10
2.2.3. Model Probabilistik.....	II- 12
2.2.3.1. <i>Binary Independence Model (BIM)</i>	II-13
2.3. <i>Relevance Feedback</i>	II-14
2.3.1. Metode Lokal	II-15
2.3.2. Metode Global	II-16
2.3.3. Algoritma Genetika.....	II- 17
2.3.3.1 Struktur Algoritma Genetika.....	II-18
BAB III METODOLOGI PENELITIAN	III-1
3.1. Identifikasi Masalah	III-1
3.2. Perumusan Masalah.....	III-2
3.3. <i>Study Literature</i>	III-2
3.4. Pengumpulan Dokumen	III-2

3.5. Analisa Sistem	III-2
3.6. Perancangan Sistem	III-4
3.7. Implementasi	III-5
3.8. Pengujian Sistem	III-5
3.9. Kesimpulan dan Saran	III-6
BAB IV ANALISA DAN PERANCANGAN	IV-1
4.1. Analisa Sistem Temu Kembali Informasi	IV-1
4.1.1. Pengumpulan Dokumen	IV-2
4.1.2. <i>Preprocessing</i>	IV-3
4.1.2.1 <i>Tokenization</i>	IV-3
4.1.2.2 <i>Filtration</i>	IV-6
4.1.2.3 <i>Stemming</i>	IV-7
4.1.2.4 Pembobotan Kata (<i>Weighting</i>)	IV-9
4.1.3. Perhitungan Nilai <i>Similarity Binary Independence Model</i>	IV-12
4.2. Analisa <i>Relevance Feedback</i> Dengan Algoritma Genetika	IV-13
4.2.1. Penentuan Populasi Solusi	IV-14
4.2.2. Penentuan Populasi Awal	IV-14
4.2.3. Mengubah Populasi Solusi Kedalam Kromosom	IV-14
4.2.4. Pembangkitan Populasi Awal Kedalam Kromosom	
Populasi Solusi	IV-14
4.2.5. Hitung Nilai <i>Fitness</i>	IV-15
4.2.6. <i>Crossover</i>	IV-15
4.2.7. <i>Mutasi</i>	IV-16
4.2.8. Pencocokan Dokumen	IV-16

4.3. Perancangan Antarmuka (<i>Interface</i>) Sistem.....	IV-17
BAB V IMPLEMENTASI DAN PENGUJIAN	V-1
5.1. Implementasi	V-1
5.1.1. Batasan Implementasi	V-1
5.1.2. Lingkungan Operasional	V-2
5.1.3. Hasil Implementasi	V-2
5.2. Pengujian	V-11
5.2.1. Rencana Pengujian	V-11
5.2.2. Lingkungan Pengujian	V-12
5.2.3. Hasil Pengujian Unjuk Kerja Sistem	V-13
5.2.4. Kesimpulan Pengujian Untuk Kerja Sistem.....	V-43
BAB VI PENUTUP	VI-1
6.1. Kesimpulan.....	VI-1
6.2. Saran	VI-2
DAFTAR PUSTAKA	xxv
LAMPIRAN	
DAFTAR RIWAYAT HIDUP	

DAFTAR TABEL

Tabel	Halaman
2.1 Parameter Menghitung <i>Precision</i> dan <i>Recall non interpolasi</i>	II-4
4.1. Hasil Proses <i>Tokenization</i>	IV-4
4.2. Hasil Proses <i>Filtration</i>	IV-6
4.3. Hasil Proses <i>Stemming</i>	IV-8
4.4. Hasil Proses <i>Weighting</i>	IV-10
4.5. Pembangkitan Kromosom Populasi Solusi	IV-14
4.6. Pembangkitan Kromosom Populasi Awal	IV-14
4.7. Pengkodean Kromosom Populasi Solusi	IV-15
4.8. Hasil <i>Fitness</i>	IV-15
5.1. Daftar <i>Query</i> Pengujian	V-13
5.2. Hasil Pengujian Interpolasi untuk <i>Query 1: Genetika</i>	V-14
5.3. Hasil Pengujian Interpolasi <i>Feedback</i> untuk No. Dokumen 2	V-15
5.4. Hasil Pengujian <i>Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q1	V-15
5.5. Hasil Pengujian <i>Feedback Precision</i> dan <i>Recall</i> <i>Non-interpolasi Pada Q1</i>	V-16
5.6. Hasil Pengujian <i>Interpolasi</i> untuk <i>Query 2: Enkripsi</i>	V-19
5.7. Hasil Pengujian <i>Interpolasi Feedback</i> untuk No. Dokumen 24	V-20
5.8. Hasil Pengujian <i>Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q2	V-21
5.9. Hasil Pengujian <i>Feedback Precision</i> dan <i>Recall</i> <i>Non-interpolasi Pada Q2</i>	V-22
5.10. Hasil Pengujian <i>Interpolasi</i> untuk <i>Query3: Kecerdasan Buatan</i>	V-24
5.11. Hasil Pengujian <i>Interpolasi Feedback</i> untuk No. Dokumen 12	V-25

5.12. Hasil Pengujian <i>Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q3.....	V-26
5.13. Hasil Pengujian <i>Feedback Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q3.....	V-27
5.14. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi Query</i> 4	V-29
5.15. Hasil Pengujian <i>Interpolasi Feedback</i> Untuk No. Dokumen 28.....	V-30
5.16. Hasil Pengujian <i>Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q4.....	V-31
5.17. Hasil Pengujian <i>Feedback Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q4.....	V-32
5.18 Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi Quer</i> 5	V-34
5.19 Hasil Pengujian <i>Interpolasi Feedback</i> Untuk No. Dokumen 25	V-35
5.20 Hasil Pengujian <i>Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q5.....	V-36
5.21 Hasil Pengujian <i>Feedback Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q5.....	V-37
5.22 Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi Quer</i> 6	V-39
5.23 Hasil Pengujian <i>Interpolasi Feedback</i> Untuk No. Dokumen 37	V-40
5.24 Hasil Pengujian <i>Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q6.....	V-41
5.25 Hasil Pengujian <i>Feedback Precision</i> dan <i>Recall Non-interpolasi</i> Pada Q5.....	V-41
5.26 Hasil Pengujian <i>Precision</i> dan <i>Recall</i>	V-44

DAFTAR GAMBAR

Gambar	Halaman
2.1. Arsitektur Sistem Temu Kembali Informasi	
(Mandala R & Setiawan H, 2006)	II-5
2.2. Contoh VSM Dengan Dua Dokumen D1 dan D2 dan Query Q1	
(Cios dkk, 2007)	II-10
2.3. Prosedur <i>Manual-Relevance Feedback</i> (Mandala,2006)	II-15
2.4. Prosedur <i>Pseude-Relevance Feedback</i> (Mandala, 2006)	II-16
3.1. Tahapan Penyusunan Tugas Akhir	III-1
4.1. Alur Proses Sistem Temu Kembali Informasi	IV-1
4.2. Alur Proses <i>Preprocessing</i>	IV-3
4.3. <i>Flowchart Tokenization</i>	IV-4
4.4. <i>Flowchart Stopword</i>	IV-6
4.5. <i>Flowchart</i> Algoritma Nazief dan Andriani	IV-8
4.6. Alur Proses <i>Relevance Feedback</i> Dengan Algoritma Genetika.....	IV-17
4.7. Rancangan Menu Utama (Menu Pencarian)	IV-18
4.8. Rancangan Hasil Pencarian Menu Utama	IV-18
4.9. Rancangan Hasil Pencarian <i>Feedback</i>	IV-19
4.10. Rancangan Menu <i>Login</i>	IV-19
4.11. Rancangan Menu Pencarian <i>Admin</i>	IV-20
4.12. Rancangan Hasil Pencarian BIM	IV-21
4.13. Rancangan Menu <i>Upload Corpus</i>	IV-21
4.14. Rancangan Menu Proses Hasil <i>Indexing</i>	IV-22

4.15. Rancangan Menu Daftar <i>Corpus</i>	IV-22
4.16. Rancangan Menu Pengujian <i>Precision</i> dan <i>Recall</i> Interpolasi	IV-23
4.17. Rancangan Menu Pengujian <i>Precision</i> dan <i>Recall</i> Non-Interpolasi	IV-23
4.18. Rancangan Menu Pengujian <i>Preprocessing</i>	IV-24
5.1. Menu Tampilan Utama (Menu Pencarian).....	V-3
5.2. Hasil Pencarian Menu Utama	V-3
5.3. Hasil Pencarian <i>Feedback</i>	V-4
5.4. Menu <i>Login</i> Sebagai <i>Admin</i>	V-4
5.5. Menu Pencarian <i>Admin</i>	V-5
5.6. Hasil Pencarian <i>Binary</i>	V-6
5.7. Hasil Pencarian <i>Feedback Admin</i>	V-6
5.8. <i>Upload Corpus</i>	V-7
5.9 <i>Indexing</i>	V-8
5.10. Hasil Detail <i>Indexing</i>	V-8
5.11. Menu Pengujian <i>Preprocessing</i>	V-9
5.12. Menu Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i>	V-10
5.13. Menu Pengujian <i>Precision</i> dan <i>Recall Non-Interpolasi</i>	V-11
5.14. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i> untuk <i>Query</i> “Genetika”	V-17
5.15. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i> untuk <i>Feedback</i> Dokumen 22	V-17
5.16. Grafik Pengujian <i>Precision</i> dan <i>Recall Non-Interpolasi</i> untuk <i>Query</i> “Genetika”	V-18
5.17. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i>	

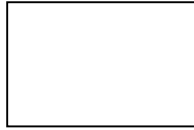
untuk <i>Query</i> “Enkripsi”	V-22
5.18. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Feedback</i> Dokumen 24	V-23
5.19. Hasil Pengujian <i>Precision</i> dan <i>Recall Non-Interpolasi</i>	
untuk <i>Query</i> “Enkripsi”	V-23
5.20. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Kecerdasan Buatan”	V-27
5.21. Hasil Pengujian <i>Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Feedback</i> Dokumen 12	V-28
5.22. Hasil Pengujian <i>Precision</i> dan <i>Recall Non-Interpolasi</i>	
Untuk <i>Query</i> ”Kecerdasan Buatan”	V-28
5.23 Grafik <i>Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Klasifikasi Keberadaan Kalimat”	V-32
5.24 Grafik <i>Feedback Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Klasifikasi Keberadaan Kalimat”	V-33
5.25 Grafik <i>Precision</i> dan <i>Recall Non-Interpolasi</i>	
untuk <i>Query</i> “Klasifikasi Keberadaan Kalimat”	V-33
5.26 Grafik <i>Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Pendeteksi Penyakit”	V-37
5.27 Grafik <i>Feedback Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Pendeteksi Penyakit”	V-38
5.28 Grafik <i>Precision</i> dan <i>Recall Non-Interpolasi</i>	
untuk <i>Query</i> “Pendeteksi Penyakit”	V-38

5.29 Grafik <i>Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Jaringan Syaraf Tiruan”	V-42
5.30 Grafik <i>Feedback Precision</i> dan <i>Recall Interpolasi</i>	
untuk <i>Query</i> “Jaringan Saraf Tiruan”	V-42
5.31 Grafik <i>Precision</i> dan <i>Recall Non-Interpolasi</i>	
untuk <i>Query</i> “Jaringan Saraf Tiruan”	V-43

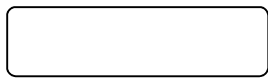
DAFTAR RUMUS

Rumus	Halaman
2.1. <i>Precision</i>	II-4
2.2. <i>Recall</i>	II-4
2.3. Rumus Untuk Menghitung <i>Precision Non-Interpolasi</i>	II-4
2.4. Rumus Untuk Menghitung <i>Recall Non-Interpolasi</i>	II-4
2.5 <i>Relevancy</i> Model Ruang Vektor	II-11
2.6 <i>Similarity</i> Model Probabilistik	II-13
2.7. <i>Similarity</i> Model <i>Binary Independence</i>	II-13
2.8 <i>Fitness</i>	II-18

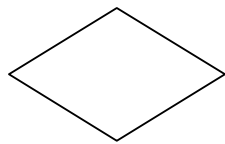
DAFTAR SIMBOL



Proses pada *flowchart*



Start/ Finish suatu proses pada
flowchart



Kondisi/ keadaan tindakan pada
flowchart berupa Ya atau Tidak

DAFTAR LAMPIRAN

Lampiran	Halaman
A. Daftar Judul Koleksi Dokumen	A-1
B. Daftar <i>Stopword</i>	B-1
C. Daftar <i>Stopcharacter</i>	C-1
D. Daftar <i>Stemming</i>	D-1

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebutuhan informasi saat ini hampir menjadi sebuah kebutuhan pokok dengan seiring perkembangan teknologi informasi. Kecepatan perubahan dan penambahan informasi menyebabkan dibutuhkannya suatu sistem yang dapat mengakses dan menyediakan berbagai informasi tersebut. Sistem dituntut bagaimana dapat menyediakan akan kebutuhan informasi yang dibutuhkan *user* dengan memberikan hasil yang akurat dan relevan. Permasalahannya bagaimana membuat sebuah sistem yang mampu memberikan hasil pencarian dokumen yang akurat dan relevan sesuai yang diinginkan pengguna.

Salah satu cara untuk mendapatkan informasi yang akurat dan relevan adalah dengan menggunakan sistem temu kembali informasi, sistem ini membuat perhitungan untuk menentukan apakah sebuah informasi relevan dengan kebutuhan penggunaanya (Ramadhany, 2008).

Sistem temu kembali informasi dapat menghasilkan informasi yang akurat dan relevan dibuat dengan menggunakan perhitungan-perhitungan tertentu atau disebut dengan model. Diantara beberapa banyak model, *Binary Independence Model* (BIM) adalah perkembangan dari model probabilistik yang digunakan untuk membangun sistem temu kembali informasi.

Binary independence model ini memiliki beberapa kelebihan yaitu berdasarkan teori perhitungan yang telah teruji dan relevansinya dimodelkan secara eksplisit. Namun model ini juga memiliki beberapa kelemahan, seperti pembobotan mengabaikan jumlah kemunculan *term*, hanya mempertimbangkan muncul atau tidak dan mengansumsikan *term* tidak saling berkaitan (Ramadhany, 2008). Kelemahan ini dapat mengurangi efektifitas kemampuan sistem dalam menghasilkan dokumen yang relevan.

Efektivitas sistem temu kembali informasi dengan *Binary independence model* diperlukan untuk mengukur kemampuan hasil *query*, yang disebut dengan

dokumen relevan sesuai dengan yang diinginkan *user*. Efektifitas tersebut dapat dilakukan dengan peningkatan kemampuan dari sistem temu kembali informasi dalam menghasilkan dokumen yang relevan sesuai dengan *query* yang dibutuhkan *user*, yaitu dengan menerapkan *relevance feedback*.

Relevance feedback adalah salah satu cara untuk meningkatkan hasil kemampuan dari kerelevanan dokumen yang dihasilkan oleh sistem temu kembali informasi. Proses *relevance feedback* akan mengembalikan dokumen hasil pencarian ke dalam sistem temu kembali informasi, kemudian sistem akan mencari dokumen lain yang memiliki tingkat kesamaan dengan dokumen yang dijadikan *feedback*. *Relevance feedback* bertujuan untuk meningkatkan *recall* (kemiripan atau kecocokan antara permintaan informasi dengan jawaban permintaan) dan *precision* (proporsi jumlah dokumen yang ditemukan kembali oleh sistem) berdasarkan informasi dari dokumen-dokumen relevan yang diidentifikasi pengguna (Adisantoso dkk, 2004).

Proses *relevance feedback* dapat dilakukan dengan banyak metode, salah satunya adalah dengan menerapkan algoritma genetika. Algoritma genetika adalah suatu algoritma pencarian berbasis pada mekanisme seleksi alam dan genetika. Algoritma ini dimulai dengan kumpulan solusi yang disebut dengan populasi. Solusi-solusi (dokumen yang dianggap relevan) dari sebuah populasi diambil dan digunakan untuk membentuk populasi yang baru. Hal ini bertujuan agar populasi yang baru dibentuk tersebut akan lebih baik dari pada yang lama. Solusi-solusi yang dipilih untuk membentuk solusi-solusi yang baru dipilih sesuai dengan *fitness* mereka masing masing. (Kusumadewi, 2003).

Berdasarkan pada penelitian sebelumnya, “*Relevance Feedback Pada Temu-Kembali Teks Berbahasa Indonesia Dengan Metode IDE-DEC-HI Dan IDE-REGULAR*” yang dilakukan Julio Adisantoso, Ahmad Ridha dan Andika Wahyu Agusetyawan (2004), menunjukkan penerapan *relevance feedback* dengan metode *IDE-DEC-HI* dan *IDE-REGULAR*, nilai *recall* meningkat mendekati 90% dan 100% sedangkan nilai *precision* meningkat 15.75% dan 12.75%, penelitian tersebut dilakukan terhadap model ruang *vector*.

Oleh karena itu, untuk meningkatkan efektifitas sistem dalam menghasilkan dokumen yang relevan dan akurat pada sistem temu kembali informasi dengan *binary independence model* dan untuk mengukur dan menerapkan sejauh mana pengaruh peningkatan kemampuan dari *relevance feedback* menggunakan algoritma genetika maka tugas akhir ini akan membahas **“Penerapan Algoritma Genetika Pada *Relevance feedback* Dalam Sistem Temu Kembali Informasi Dengan Menggunakan *Binary Independence Model*”**.

1.2 Rumusan Masalah

Berdasarkan dari latar belakang permasalahan di atas, maka dibuat perumusan masalah yaitu bagaimana menerapkan algoritma genetika pada *relevance feedback* dalam sistem temu kembali informasi dengan menggunakan *binary independence model* dan sejauh mana peningkatan peformansi dari kerelevanan dokumen yang dihasilkan oleh sistem temu kembali informasi tersebut.

1.3 Batasan Masalah

Adapun yang menjadi batasan masalah dalam laporan tugas akhir ini adalah:

1. Jumlah dokumen yang digunakan dalam penelitian sebanyak 100 dokumen. Koleksi dokumen (*corpus*) yang digunakan adalah jurnal-jurnal informatika, hanya yang berekstensi *.txt*.
2. Mekanisme penggunaan *relevance feedback* dilakukan dengan jenis *Manual relevance feedback* dan menggunakan algoritma genetika.
3. Tidak menganalisa terhadap proses *stemming* dan tidak melakukan pengujian *stemming* terhadap hasil keluarannya, hanya menggunakan algoritma *stemming* yang sudah ada, algoritma *stemming* yang digunakan adalah Algoritma Nazief dan Adriani.

1.4 Tujuan

Tujuan dari laporan tugas akhir ini adalah sebagai berikut:

1. Dapat menerapkan algoritma genetika pada *relevance feedback* dalam sistem temu kembali informasi yang menggunakan metode *binary independence*.
2. Untuk mengukur peningkatan kemampuan dari kerelevanan dokumen yang dihasilkan oleh sistem temu kembali informasi dengan penerapan *relevance feedback* menggunakan algoritma genetika.
3. Membandingkan tingkat *precision* dan *recall* dari hasil pencarian menggunakan metode *binary independence* dengan hasil pencarian *feedback* penerapan algoritma genetika.

1.5 Sistematika Pembahasan

Sistematika penulisan laporan Tugas Akhir ini terdiri dari pokok-pokok permasalahan yang dibahas pada masing-masing yang diuraikan menjadi beberapa bagian:

BAB I. PENDAHULUAN

Bab ini membahas tentang gambaran umum isi tugas akhir yang meliputi latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan sistematika penulisan.

BAB II. LANDASAN TEORI

Bab ini menjelaskan tentang teori-teori yang berhubungan dengan penelitian yang terdiri dari penjelasan mengenai konsep sistem temu kembali informasi, pengukuran performa sistem temu kembali informasi, arsitektur sistem temu kembali informasi, koleksi dokumen, model *boolean*, model *vector*, model probabilistik, *binary independence model*, *relevance feedback*, dan Algoritma genetika.

BAB III. METODOLOGI PENELITIAN

Bab ini membahas langkah-langkah yang dilaksanakan dalam proses penelitian, yaitu identifikasi masalah, perumusan masalah, *study literature*, analisa sistem, perancangan sistem, implementasi, pengujian sistem, dan kesimpulan dan saran.

BAB IV. ANALISA DAN PERANCANGAN

Berisi pembahasan mengenai analisa sistem temu kembali, analisa *relevance feedback* dengan algoritma genetika, contoh implementasi pencarian dokumen, dan perancangan antarmuka (*interface*) sistem.

BAB V. IMPLEMENTASI DAN PENGUJIAN

Dalam bab ini membahas tentang implementasi dari sistem temu kembali informasi yang meliputi batasan implementasi, lingkungan operasional dan hasil implementasi. Kemudian membahas mengenai pengujian dari sistem, yaitu meliputi lingkungan pengujian, hasil pengujian untuk sistem, dan kesimpulan pengujian kerja sistem.

BAB VI. PENUTUP

Dalam bab ini akan membahas kesimpulan dari hasil penelitian mengenai ini dan saran untuk menyempurnakan penelitian ini untuk berikutnya.

BAB II

LANDASAN TEORI

Sistem informasi didefinisikan oleh James O'Brien adalah suatu kombinasi yang terorganisir tentang orang-orang, perangkat keras, perangkat lunak jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah bentuk, dan menghamburkan informasi di dalam suatu organisasi (Trunojoyo, 2010)

Informasi merupakan sebuah data yang telah diolah dan diproses sedemikian rupa sehingga memiliki arti dan dapat berupa fakta atau sebuah nilai yang bermanfaat. Informasi antara lain dapat berupa dokumen, surat, berita, gambar bahkan sebuah video. Informasi dapat diakses dari dua media, media cetak dan media elektronik (Wakid dkk, 2011).

2.1 Sistem Temu Kembali Informasi

Standar ISO 2382/1, mendefinisikan *Information Retrieval* (IR) sebagai tindakan, metode dan prosedur untuk menemukan kembali data yang tersimpan, kemudian menyediakan informasi mengenai subyek yang dibutuhkan. Tindakan tersebut mencakup *text indexing*, *inquiry analysis*, dan *relevance analysis*. Data mencakup teks, tabel, gambar, ucapan, video, dan lainnya serta informasi termasuk pengetahuan terkait yang dibutuhkan untuk mendukung penyelesaian masalah dan akuisisi pengetahuan (Cios dkk, 2007).

2.1.1 Pengertian Sistem Temu Kembali Informasi

Ada beberapa ahli mendefinisikan sistem temu kembali sebagai berikut:

1. Meadow (1992:2)

Sistem temu kembali informasi adalah sebuah proses komunikasi, dikatakan sebuah proses komunikasi berate penulis berkomunikasi dengan pembaca, tetapi tidak langsung amun berbeda waktu yang mungkin panjang antara penciptaan pesan kepada pengguna sistem.

2. Harter (1986:2)

Sistem temu kembali informasi merupakan seperangkat hubungan antara pengguna potensial informasi dengan pengumpul informasi itu sendiri. Dengan tujuan untuk memberi informasi yang dibutuhkan dan menyaring informasi yang kurang relevan atau tidak sesuai dengan kebutuhan pengguna.

3. Sulistiyo Basuki

Sistem temu kembali informasi yaitu sejumlah kegiatan yang bertujuan menyediakan dan memasok informasi bagi pengguna sebagai jawaban atas permintaan pemakai. Temu balik informasi (*information retrieval*) merujuk keseluruhan kegiatan yang meliputi pembuatan wakil informasi (*representation*), penyimpanan (*storage*), pengaturan (*organization*) sampai ke pengambilan (*access*).

4. Salton dalam Hasugian (1989)

STBI suatu proses untuk mengidentifikasi, mengenali dan memanggil dokumen tertentu dalam rangka memberikan jawaban atas permintaan informasi. Dari pendapat Salton diatas dapat disimpulkan bahwa, "Terpanggil tidaknya suatu dokumen tergantung dengan kesamaan *Query* dengan wakil dokumen".

2.1.2 Istilah Penting Dalam Sistem Temu Kembali Informasi

Ada beberapa istilah penting yang berkaitan dengan sistem temu kembali informasi. Istilah tersebut antara lain (Ramadhany, 2008):

1. *Query*

Query adalah ekspresi kebutuhan akan informasi. *Query* biasanya berupa kata kunci yang diberikan kepada sistem untuk mendapatkan informasi yang relevan terhadap kebutuhan pada *query*.

2. Dokumen

Dokumen adalah sebuah objek yang mengandung informasi yang akan dieksplorasi oleh sistem temu kembali informasi.

3. Relevansi

Relevansi adalah tingkat kesesuaian sebuah dokumen dengan *query* yang diberikan oleh pengguna. Relevansi dihitung oleh rumus tertentu tergantung pada model yang digunakan dalam sistem temu kembali informasi.

4. Rangking

Rangking adalah pengurutan dokumen-dokumen yang telah dicari oleh sistem temu kembali informasi. Acuan dalam pengurutan ini adalah nilai relevansi tiap dokumen terkait *query* yang diberikan.

5. Term

Term adalah kata yang memiliki arti yang terdapat pada dokumen dan *query*. Dari semua kata yang ada, kata-kata yang merupakan kata umum dihilangkan sehingga yang tersisa hanya kata yang benar-benar berhubungan dengan isi dokumen. Kata yang tersisa tersebut yang disebut *term*.

2.1.3 Pengukuran Kemampuan Dalam Sistem Temu Kembali Informasi

Sistem temu kembali informasi (*information retrieval system*) digunakan untuk menemukan kembali (*retrieve*) informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis (Mandala R & Setiawan H, 2002). Tujuan dari sistem temu kembali informasi yang ideal adalah:

1. Menemukan seluruh dokumen yang relevan terhadap suatu *query*.
2. Hanya menemukan dokumen relevan saja, artinya tidak terdapat dokumen yang tidak relevan pada dokumen hasil pencarian.

Sistem temu kembali informasi memiliki beberapa cara untuk mengukur kemampuan dalam memerikan hasil pencarian. Pengukuran ini tergantung dari kumpulan dokumen dan *query* terhadap dokumen yang telah diketahui relevansi sebelumnya. Cara mengukur kinerja itu dengan *precision* dan *recall* secara *interpolasi* dan *precision* dan *recall* secara *non interpolasi*, perhitungan interpolasinya sebagai berikut (Ramadhany, 2008):

1. Precision

Precision adalah perbandingan dokumen relevan yang ditemukan terhadap semua dokumen yang berhasil ditemukan.

$$\textbf{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \dots\dots\dots(2.1)$$

2. Recall

Recall adalah perbandingan dokumen relevan yang ditemukan terhadap semua dokumen relevan yang ada.

$$\textbf{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \dots\dots\dots(2.2)$$

Sedangkan untuk menghitung *precision* dan *recall* secara *non interpolasi* seperti pada tabel 2.1 berikut:

Tabel 2.1 Parameter Menghitung *Precision* dan *Recall non interpolasi*

Keterangan	Relevan	Tidak relevan
Terambil	<i>True positive (tp)</i>	<i>False positive (fp)</i>
Tidak terambil	<i>False negative (fn)</i>	<i>True negative (tn)</i>

Rumus untuk menghitung *Precision Non-Interpolasi*:

$$P = \frac{tp}{tp + fp} \dots\dots\dots(2.3)$$

Keterangan :

$$P = \textit{Precision} \qquad tp = \textit{true positive} \qquad fp = \textit{false positive}$$

Rumus untuk menghitung *Recall Non-Interpolasi*:

$$R = \frac{tp}{tp + fn} \dots\dots\dots(2.4)$$

Keterangan :

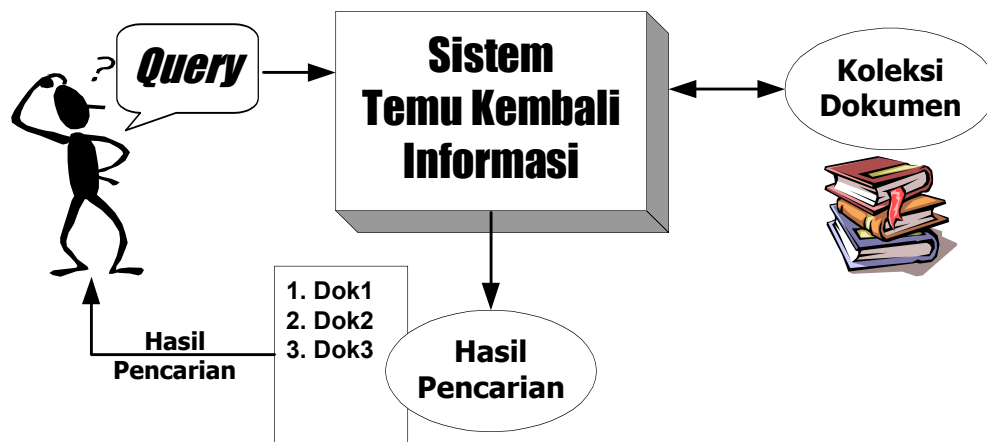
$$R = \textit{Recall} \qquad tp = \textit{true positive} \qquad fn = \textit{false negative}$$

2.1.4 Arsitektur Sistem Temu Kembali Informasi.

Ada dua pekerjaan yang ditangani oleh sistem ini, yaitu melakukan *pre-processing* terhadap database dan kemudian menerapkan metode tertentu untuk menghitung kedekatan (relevansi atau *similarity*) antara dokumen di dalam database yang telah di-*preprocess* dengan *query* pengguna. Pada tahapan *preprocessing*, sistem yang berurusan dengan dokumen *semi-structured* biasanya memberikan *tag* tertentu pada *term-term* atau bagian dari dokumen; sedangkan pada dokumen tidak terstruktur proses ini dilewati dan membiarkan *term* tanpa

imbuhan *tag*. *Query* yang dimasukkan pengguna dikonversi sesuai aturan tertentu untuk mengekstrak *term-term* penting yang sejalan dengan *term-term* yang sebelumnya telah diekstrak dari dokumen dan menghitung relevansi antara *query* dan dokumen berdasarkan pada *term-term* tersebut. Sebagai hasilnya, sistem mengembalikan suatu daftar dokumen terurut *descending (ranking)* sesuai nilai kemiripannya dengan *query* pengguna (Cios dkk, 2007).

Arsitektur sistem temu balik informasi dapat dilihat pada gambar 2.1 (Mandala R & Setiawan H, 2006):



Gambar 2.1. Arsitektur Sistem TemuKembali Informasi
(Mandala R & Setiawan H, 2006)

2.1.5 Koleksi Dokumen (*Corpus*)

Istilah *corpus* pada prinsipnya bermakna koleksi dokumen yang di *index* dan dijadikan target pencarian. Suatu *corpus* modern memiliki beberapa karakteristik yakni: (Sinclair, 2004)

1. Modul teks, apakah bahasa berasal dalam pidato atau tulisan, atau mungkin saat dalam mode elektronik.
2. Jenis teks, misalnya jika ditulis, baik buku, jurnal, pemberitahuan atau surat.
3. Domain dari teks, misalnya apakah akademis atau populer.
4. Varietas bahasa atau bahasa yang digunakan dalam *corpus*.

5. Letak teks, misalnya (bahasa Inggris dari) Inggris atau Australia.
6. Tanggal teks.

Suatu *corpus* pengujian sistem temu kembali informasi terdiri dari:

1. Koleksi Dokumen
2. Topik-topik, yang dapat digunakan sebagai *query*
3. *Relevance judgement*, sebagai daftar dokumen yang relevan dengan topik-topik yang tersedia.

Secara garis besar proses tahapan *indexing* dalam sistem temu kembali adalah pembangunan *indexing*. Pembangunan *index* dari koleksi dokumen merupakan tugas pokok pada tahapan *Preprocessing* didalam sistem temu kembali informasi. Kualitas *index* mempengaruhi efektivitas dan efisiensi sistem temu kembali informasi. *Index* dokumen adalah himpunan *term* yang menunjukkan isi atau topik yang dikandung oleh dokumen.

Index akan membedakan suatu dokumen dari dokumen lain yang berada di dalam koleksi. Ukuran *index* yang kecil dapat memberikan hasil buruk dan mungkin beberapa item yang relevan terabaikan. *Index* yang besar memungkinkan ditemukan banyak dokumen yang relevan tetapi sekaligus dapat menaikkan jumlah dokumen yang tidak relevan dan menurunkan kecepatan pencarian (Trunojoyo, 2010).

Terdapat lima langkah pembangunan *inverted index* (Trunojoyo, 2010), yaitu:

1. Penghapusan format dan markup dari dalam dokumen

Tahap ini menghapus semua *tag* markup dan format khusus dari dokumen, terutama pada dokumen yang mempunyai banyak *tag* dan format seperti dokumen (X) HTML.

2. Pemisahan rangkaian kata (*tokenization*)

Tokenization adalah tugas memisahkan deretan kata di dalam kalimat, paragraf atau halaman menjadi *token* atau potongan kata tunggal atau *termed word*. Tahapan ini juga menghilangkan karakter-karakter tertentu seperti tanda baca dan mengubah semua token ke bentuk huruf kecil (*lower case*).

3. Penyaringan (*filtration*)

Pada tahapan ini ditentukan *term* mana yang akan digunakan untuk merepresentasikan dokumen sehingga dapat mendeskripsikan isi dokumen dan membedakan dokumen tersebut dari dokumen lain di dalam koleksi.

Term yang sering digunakan dianggap sebagai *stop-word* dan dihapus. Penghapusan *stop-word* dari dalam suatu koleksi dokumen pada satu waktu membutuhkan banyak waktu. Solusinya adalah dengan menyusun suatu pustaka *stop-word* atau *stop-list* dari *term* yang akan dihapus (Manning, 2009).

4. Konversi *term* ke bentuk akar (*stemming*)

Stemming adalah proses konversi *term* ke bentuk umumnya. Dokumen dapat pula diekspansi dengan mencari sinonim bagi *term-term* tertentu di dalamnya. Seperti *stemming*, operasi ini bertujuan menemukan suatu kelompok kata terkait. Akan tetapi sinonim bekerja berdasarkan pada *thesaurus*, tidak berbagi-pakai *term stem* (Cios dkk, 2007).

Mengubah kedalam bentuk kata dasar dapat dilakukan dengan menghilangkan akhiran atau awalan dari sebuah kata. Karena *stemming* menghilangkan imbuhan dari sebuah kata dan tiap bahasa memiliki cara tersendiri dalam menambahkan imbuhan di dalamnya, maka algoritma *stemming* yang dipakai pun harus sesuai dengan bahasa dari artikel atau dokumen yang akan diproses. Dengan proses *stemming*, jumlah ragam kata yang ada di dalam artikel ataupun dokumen dapat berkurang dan dapat mengoptimalkan *preprocessing* sehingga hasilnya lebih memuaskan (Trunojoyo, 2010).

Ada tiga jenis metode *stemming*, antara lain (Trunojoyo, 2010):

a. *Successor Variety (SV)*:

Yaitu lebih mengutamakan penyusunan huruf dalam kata dibandingkan dengan pertimbangan atas fonem. Contoh untuk kata-kata : *corpus*, *able*, *axle*, *accident*, *ape*, *abaout* menghasilkan SV untuk kata *apple*:

Karena huruf pertama dari kata “*apple*” adalah “a”, maka kumpulan kata yang ada substring “a” diikuti “b”, “x”, “c”, “p”, disebut SV dari “a” sehingga “a” memiliki 4 SV. Karena dua huruf pertama dari kata “*apple*” adalah “ap”. Maka kumpulan kata yang ada substring “ap” hanya diikuti “e” disebut SV dari “ap” sehingga “ap” memiliki 1 SV.

b. *N-Gram Conflation*:

Yaitu ide dasarnya adalah pengelompokan kata-kata secara bersama berdasarkan karakter-karakter (*substring*) yang teridentifikasi sepanjang N karakter.

c. *Affix Reemoval* (penghilangan Imbuhan):

Yaitu membuang *prefix* (awalan) dan *suffix* (akhiran) dari term menjadi suatu *term*.

5. Pemberian bobot terhadap *term* (*weighting*)

Setiap *term* diberikan bobot sesuai dengan skema pembobotan yang dipilih, apakah pembobotan lokal, global atau kombinasi keduanya. Salah satu cara untuk memberi bobot terhadap suatu kata adalah memberikan nilai jumlah kemunculan suatu kata (*term frequency*) sebagai bobot. Semakin besar kemunculan suatu kata dalam dokumen akan memberikan nilai kesesuaian yang semakin besar.

Faktor lain yang diperhatikan dalam pemberian bobot adalah kejarang munculan kata (*term scarcity*) dalam koleksi. Kata yang muncul pada sedikit dokumen harus dipandang sebagai kata yang lebih penting (*uncommon terms*) dari pada kata yang muncul pada banyak dokumen. Pembobotan akan memperhitungkan faktor kebalikan frekuensi dokumen yang mengandung suatu kata (*inverse document frequency*) (Mandala R & Setiawan H, 2002).

2.2 Model-Model Sistem Temu Kembali Informasi

Untuk mendapatkan hasil yang maksimal dalam pencarian informasi, maka banyak macam perhitungan yang digunakan untuk mengukur relevansi. Ada banyak model berbasis matematis yang tersedia dan bisa digunakan pada sistem temu kembali informasi. Secara umum, model tersebut bisa dibagi tiga, yaitu (Ramadhany, 2008):

1. Model *set-theoretic*

Model pada kategori ini mempresentasikan dokumen yang ada dalam himpunan-himpunan dokumen. Model *set-theoretic* yang umum digunakan adalah model *boolean*.

2. Model aljabar

Model pada kategori ini mempresentasikan dokumen yang ada kedalam *vector*, matrik atau tuple. Model aljabar yang umum digunakan adalah model ruang *vector*.

3. Model probabilistik

Pada model probabilistik dokumen diproses dengan berbagai langkah matematis. Similaritas kemudian dipersentasikan sebagai kemungkinan-kemungkinan. Model probabilistik yang banyak digunakan adalah *binaray independence*.

2.2.1 Model *Boolean*

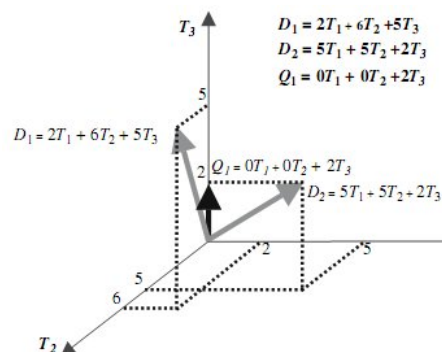
Model *boolean* adalah model yang paling awal dikenal dan paling mudah untuk diimplementasikan. Hanya saja, model *boolean* tidak dapat memberikan hasil yang diharapkan dan sangat lambat dalam *run-time* (Jaya, 2007). Model *boolean* merupakan salah satu model dimana proses pencarian informasi dari *query* yang diterima diperlakukan dengan ekspresi *boolean*. Ekspresi *boolean* yang dimaksud dapat berupa operator logika AND, OR, dan NOT. Dokumen yang di-*retrieve* adalah dokumen yang benar-benar sesuai dengan *query*.

Beberapa karakteristik dari model *boolean* dalam sistem temu kembali adalah:

1. Model *boolean* merupakan model sederhana yang menggunakan teori dasar himpunan sehingga mudah diimplementasikan.
2. Model *boolean* tidak menggunakan peringkat dokumen yang terambil. Dokumen yang terambil hanya dokumen yang benar-benar sesuai dengan pernyataan *boolean/query* yang diberikan. Sehingga dokumen yang terambil bisa sangat banyak atau bisa sedikit. Akibatnya ada kesulitan dalam mengambil keputusan.
3. Teori himpunan memang mudah, namun tidak demikian halnya dengan pernyataan *boolean* yang bisa kompleks. Akibatnya pengguna harus memiliki pengetahuan banyak mengenai kueri dengan *boolean* agar pencarian menjadi efisien.

2.2.2 Model Vektor

Dalam sistem IR, kemiripan antar dokumen didefinisikan berdasarkan representasi *bag of words* dan dikonversikan ke suatu model ruang vektor (*vector space model-VSM*). Pada VSM, setiap dokumen didalam *database* dan *query* pengguna direpresentasikan oleh suatu vektor multi-dimensi seperti yang ditunjukkan oleh Gambar 2.3.



Gambar 2.2. Contoh VSM dengan Dua Dokumen D_1 dan D_2 , dan Query Q_1 (Sumber: Cios dkk, 2007)

Berdasarkan gambar 2.2, dapat diketahui bahwa sudut yang dibentuk antara Q_1 dan D_1 lebih kecil daripada Q_1 dan D_2 . Perhitungan persamaan antara vektor *Query* dan vektor dokumen dilihat dari sudut yang terkecil, yaitu antara Q_1 dan D_1 . Sudut yang dibentuk oleh dua vektor ini dapat dihitung dengan melakukan perkalian dalam (*inner product*), sehingga rumus relevansinya adalah:

$$R(Q, D) = \cos \theta = \frac{Q \cdot D}{\|Q\| \|D\|} \dots \dots \dots (2.5)$$

Berbeda dengan model *boolean* yang menggunakan nilai biner sebagai bobot *index term*, VSM melakukan pembobotan berdasarkan *term* yang sering muncul dalam dokumen atau dikenal dengan sebutan *term frequency (tf)* dan jumlah kemunculannya dalam koleksi dokumen yang disebut *inverse document frequency (idf)* (Manning dkk, 2009).

Pada model ruang vektor, pembobotan terhadap *term* dilakukan dengan mengalikan bobot lokal *tf* dan bobot global *idf*, dikenal dengan pembobotan *tf-idf*. Metode pembobotan ini dilakukan dengan memberikan bobot kepada *term* yang penting. Artinya, *term* yang jika muncul di suatu dokumen maka, dokumen tersebut dapat dianggap relevan dengan *query* pengguna.

Model ruang vektor tidak membutuhkan komputasi yang berlebihan sehingga waktu untuk mengeksekusi akan semakin cepat dan lebih efektif (Ramadhany, 2008).

Beberapa karakteristik dari model vektor dalam sistem temu kembali adalah:

1. Model vektor berdasarkan *keyterm*.
2. Model vektor mendukung *partial matching* (sebagian sesuai) dan penentuan peringkat dokumen.
3. Prinsip dasar vektor model adalah sebagai berikut:
 - a. Dokumen direpresentasikan dengan menggunakan vektor *keyterm*
 - b. Ruang dimensi ditentukan oleh *keyterms*

- c. *Query* direpresentasikan dengan menggunakan vektor *keyterm*
 - d. Kesamaan *document-keyterm* dihitung berdasarkan jarak vector
4. Model vektor memerlukan:
- a. Bobot *keyterm* untuk vektor dokumen
 - b. Bobot *keyterm* untuk *query*
 - c. Perhitungan jarak untuk vektor *document keyterm*
5. Kinerja
- a. Efisien
 - b. Mudah dalam representasi
 - c. Dapat diimplementasikan pada *document matching*

Prosedur model ruang vektor dapat dikelompokkan menjadi tiga tahap yaitu:

1. Pengindeks-an dokumen
2. Pembobotan indeks, untuk menghasilkan dokumen yang relevan
3. Memberikan peringkat dokumen berdasarkan ukuran kesamaan (*similarity measure*).

Model ini berhasil memberikan hasil yang lebih baik dibandingkan model *boolean*. Model ini juga dapat menampilkan hasil temu balik secara terurut (Jaya, 2007).

2.2.3 Model Probabilistik

Sistem temu kembali informasi juga memperkenalkan model probalistik. Model ini mengurutkan dokumen dalam urutan menurun terhadap peluang relevansi sebuah dokumen pada informasi yang dibutuhkan (Ramadhany, 2008). Beberapa model yang juga dikembangkan berdasarkan perhitungan probabilistik yaitu, *Binary Independence Model*, model Okapi BM25, dan *Bayesian Network Model* (Manning dkk, 2009). Dalam model probabilistik dasar, kemiripan (*similarity*) sebuah dokumen terhadap *Query* dihitung dengan menggunakan rumus seperti pada Persamaan 2.6.

$$RSV(d) = \sum_{t \in q} \log \frac{(c_{dt} + 1) / (c_d + 1)}{(c_{qt} + 1) / (c_q + 1)} \dots\dots\dots(2.6)$$

Keterangan:

RSV = *retrieval status value* (nilai untuk perankingan dokumen).

s = jumlah dokumen yang relevan yang mengandung *term t* pada *query q*.

S = jumlah dokumen yang relevan untuk *query q*.

df_t = jumlah dokumen dalam *corpus* yang mengandung *term t* pada *query q*.

N = jumlah dokumen dalam *corpus*.

2.2.3.1 Binary Independence Model (BIM)

Pada *binary independence model*, tiap dokumen dianggap sebagai sebuah *vector term*. Jika pada model lain jumlah kemunculan (*term frequency*) sebuah *term* diperhitungkan, maka pada *binary independence model* nilainya berupa biner, yaitu ada atau tidak ada. Bagian probabilistik dari model ini terdapat pada perhitungan similaritas sebuah dokumen dengan *query*. Sedangkan bagian *independence* pada *binary independence model* adalah karena model ini menganggap kata yang muncul pada sebuah dokumen tidak saling berkaitan (Ramadhany, 2008). Rumus relevansi *Binary Independence model*:

$$\text{Similarity} = \left| \sum_{k=1}^n w_k \cdot \log \frac{(c_{dk} + 1)}{(c_k + 1)} \right| \dots\dots\dots(2.7)$$

Keterangan:

k = Kata yang ada dalam *query*

n = Jumlah kata pada *query*

w_k = Bobot biner dari kata tersebut terhadap dokumen *query*

p_k = Konstanta sebuah dokumen yang relevan mengandung kata pada *query*

q_k = Nilai sebuah dokumen yang tidak relevan mengandung kata pada *query*

Pada inisialisasi nilai p_k ini tidak diketahui dan biasanya diisi dengan angka 0.5. Sedangkan q_k didapatkan dari perhitungan n_i/N , dengan n_i adalah

jumlah dokumen yang mengandung *term* i , sedangkan N adalah jumlah total dokumen dalam koleksi (Ramadhany, 2008).

Pada *binary independence model* memiliki beberapa kelebihan yaitu berdasarkan teori perhitungan yang telah teruji dan relevansinya dimodelkan secara eksplisit. Namun kekurangan model ini adalah definisi awal dari R (jumlah dokumen yang relevan dengan *query*) harus diperkirakan, pembobotan mengabaikan jumlah kemunculan *term* (hanya mempertimbangkan muncul atau tidak), dan mengasumsikan *term* tidak saling berkaitan (Ramadhany, 2008).

2.3 *Relevance Feedback*

Relevance feedback adalah salah satu cara untuk meningkatkan hasil kinerja dari kerelevanan dokumen yang dihasilkan oleh sistem temu kembali informasi. Ide dasarnya adalah dengan melakukan sebuah *query*, kemudian menerima umpan balik dari pengguna mengenai dokumen yang relevan dengan *query* yang diberikan tadi. Kata-kata dari dokumen yang dianggap relevan tersebut kemudian ditambahkan ke *query* (Ramadhany, 2008).

Relevance feedback bertujuan untuk meningkatkan *recall* dan *precision* berdasarkan informasi dari dokumen-dokumen relevan yang diidentifikasi pengguna (Adisantoso, 2004). *Relevance feedback* adalah interaksi antara pengguna dan sistem untuk secara bersama-sama merundingkan masalah *query* yang tepat untuk menggambarkan kebutuhan informasi. Proses *relevance feedback* akan mengubah *query* awal menjadi *query* baru yang menggambarkan lebih jelas mengenai kebutuhan informasi yang dibutuhkan oleh pengguna.

Dalam penentuan *relevance feedback* oleh pengguna dimaksudkan untuk mencari dokumen lanjut selain dari dokumen yang telah ditemukan tersebut. Apakah dengan proses ini dokumen lanjut selain dari dokumen yang telah ditemukan tersebut. Apakah dengan proses ini akan ditemukan dokumen lain yang relevan atau tidak. Proses temu kembali informasi dengan proses *relevance feedback* yang baik akan menemukan dokumen-dokumen lain yang memiliki relevansi dengan *query*.

Pencarian dokumen dilakukan dengan penambahan *term/kata* pada *query* sebelumnya sesuai dengan proses dari *relevance feedback*. Sedangkan pencarian dokumen dilakukan pada selain dokumen yang telah ditemukan tersebut (Adisantoso, 2004).

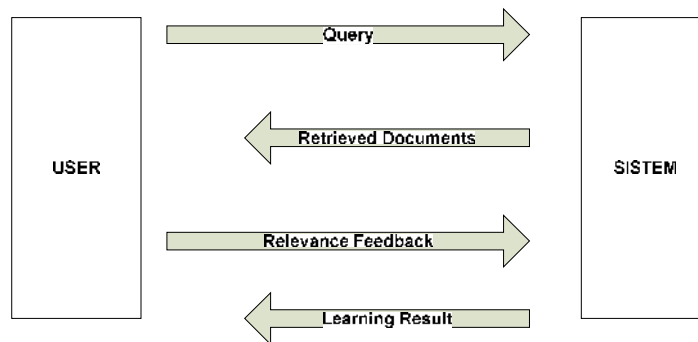
Ada beberapa metode untuk *relevance feedback*, yaitu metode lokal dan metode global (Adisantoso, 2004).

2.3.1 Metode Lokal

Ide dari metode lokal adalah memperluas *query* awal berdasarkan informasi yang didapat dari beberapa dokumen urutan teratas yang diambil pertama kali oleh sistem.

Metode ini sebenarnya ada dua jenis, *manual-relevance feedback* (umpan balik yang dilakukan oleh pengguna) dan *pseudo-relevance feedback/Automatic relevance feedback* (umpan balik yang dilakukan oleh sistem).

Pada *manual-relevance feedback* sistem menyodorkan beberapa dokumen hasil pencarian pertama. Pengguna memberikan tanda dokumen mana saja yang relevan, kemudian mengembalikannya kembali kepada sistem dari hasil informasi dokumen relevan tersebut, sistem akan memperluas *query* asal dan melakukan pencarian ulang.



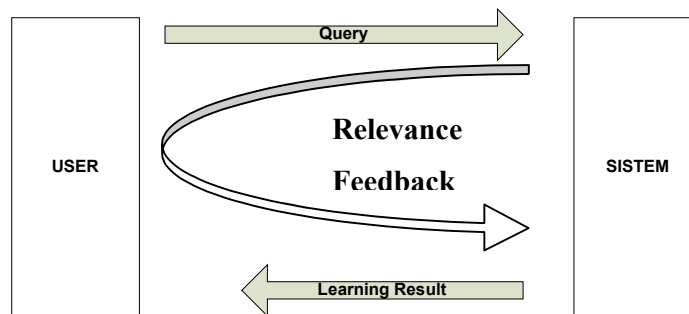
Gambar 2.3 Proses *Manual-Relevance Feedback*

Manual-relevance feedback melakukan 5 buah proses utama, yaitu (Mandala, 2006):

- a. Inisialisasi pencarian dokumen.
- b. Memberikan hasilnya kepada pengguna.

- c. Menerima umpan balik dari pengguna.
- d. Membuat *query* baru berdasarkan umpan balik dan melakukan pencarian ulang.
- e. Memberikan hasil pencarian ulang kepada pengguna.

Sedangkan pada *pseudo-relevance feedback*/*Automatic relevance feedback* merupakan cara untuk mengurangi gangguan terhadap pengguna. Dalam cara ini sistem tidak langsung menampilkan dokumen-dokumen hasil pencarian pertama, tetapi sistem mengambil beberapa dokumen dengan urutan teratas dari hasil pencarian pertama dan menggunakannya untuk memperluas *query* dengan menganggap bahwa dokumen tersebut relevan.



Gambar 2.4 Proses Pseude-Relevance Feedback

Proses-proses dalam *pseudo-relevance feedback* adalah sebagai berikut (Mandala, 2006):

- a. Inisialisasi pencarian dokumen.
- b. N-dokumen pertama yang ditemukan digunakan sebagai umpan balik.
- c. Membuat *query* baru dari umpan balik dan melakukan pencarian ulang.
- d. Memberikan hasil pencarian kepada pengguna.

2.3.2 Metode global

Berbeda dengan metode lokal, sistem dengan metode global melakukan ekspansi *query* terlebih dahulu sebelum dilakukan *retrieval*. Dua alur pemrosesan pada analisis lokal tetap ada, namun diawal sekali terdapat sebuah proses yang akan menghasilkan basis data kata benda yang nantinya akan digunakan untuk ekspansi *query*. Pembuatan basis data kata benda didasarkan pada seringnya

kemunculan sebuah kata benda dengan benda lain untuk mendefinisikan sebuah konsep. Semakin sering muncul kata benda dengan sebuah kata benda tertentu maka akan semakin tinggi nilainya. Disinilah letak perbedaan metode ekspansi analisis lokal dan global. Sedangkan kedua jalur pemrosesannya sama saja. Pada sistem ini ada tambahan masukan, yaitu *file* yang berisi *lexicon* yang akan digunakan untuk membentuk basis data kata benda.

Lexicon adalah semacam kamus bahasa yang memberikan informasi jenis kata (kata benda, kata sifat, kata keterangan). Proses inilah yang disebut *noun phrase parsing*. Salah satu metode global yang terkenal adalah *thesaurus*. *Thesaurus* menyediakan informasi berdasarkan sinonim dan kata-kata yang saling berhubungan serta frase-frase.

Thesaurus dapat menambah *recall* tetapi secara signifikan dapat mengurangi *precision*, terutama dengan kata-kata yang ambigu. Keuntungan dari metode ini adalah *robust*, basis data yang tercipta dapat digunakan berulang kali untuk *query* yang berbeda. Sedangkan kerugiannya adalah metode ini memakan tempat (*disk space*) dan perlu waktu cukup lama untuk membangun basis data konsepnya. Secara keseluruhan, metode global tidak sebaik *relevance feedback* tetapi sama baiknya dengan *pseudo relevance feedback*.

2.3.3 Algoritma Genetika

Algoritma genetika adalah suatu algoritma pencarian berbasis pada mekanisme seleksi alam dan genetika. Algoritma genetika merupakan salah satu algoritma yang sangat tepat digunakan dalam menyelesaikan optimasi kompleks, yang sulit dilakukan oleh metode konvensional. Algoritma ini dimulai dengan kumpulan solusi yang disebut dengan populasi. Solusi solusi dari sebuah populasi diambil dan digunakan untuk membentuk populasi yang baru. Hal ini dimotivasi dengan harapan bahwa populasi yang baru dibentuk tersebut akan lebih baik daripada yang lama. Solusi solusi yang dipilih untuk membentuk solusi solusi yang baru dipilih sesuai dengan *fitness* mereka masing masing. (Kusumadewi, 2003).

2.3.3.1 Struktur Algoritma Genetika

Algoritma genetika memiliki karakteristik-karakteristik yang dilakukan dalam pemrosesannya. Berikut tahapan dalam proses algoritma genetika adalah sebagai berikut:

a. Populasi Solusi

Populasi solusi adalah populasi yang diambil dari *query* atau dokumen yang dijadikan dasar pencarian.

b. Populasi Awal

Populasi awal adalah seluruh dokumen yang akan dijadikan objek pencarian (solusi yang mungkin diambil). Dimana setiap populasi akan direpresentasikan dengan nilai biner (1 atau 0)

c. *Fitness*

Nilai *fitness* adalah nilai yang menyatakan baik tidaknya suatu solusi (individu), yang dijadikan acuan dalam mencapai nilai optimal dalam algoritma genetika (Basuki, 2003).

Nilai *fitness* banyak cara dalam proses perhitungannya, salah satu caranya adalah dengan metode perhitungan Hsinchun Chen, Menurut (Chen, 1992) ukuran kemiripan dengan Hsinchun Chen adalah setiap istilah dapat diwakili oleh satu simpul (node), dimana hubungan antar istilah. Berdasarkan penelitian sebelumnya “*Information Retrieval System Dengan Menggunakan Algoritma Genetika Pada Tugas Akhir*” yang dilakukan oleh Anhar A. S, (2012). Perhitungannya dengan menggunakan rumus persamaan 2.8 yaitu sebagai berikut:

$$\overline{\text{---}} \quad \dots\dots\dots (2.8)$$

d_{ijk} = gen dari kromosom populasi yang bernilai 1 pada pengkodean kromosom populasi.

di_j = gen dari kromosom solusi yang direpresentasikan dengan nilai 1.

d. *Crossover*

Operator persilangan merupakan operasi yang bekerja untuk menggabungkan dua kromosom orang tua (*parent*) menjadi kromosom baru.

Beberapa jenis *crossover* tersebut adalah:

a. Crossover satu titik

Proses *crossover* dilakukan dengan memisahkan suatu *string* menjadi dua bagian dan selanjutnya salah satu bagian dipertukarkan dengan salah satu bagian dari string yang lain yang telah dipisahkan dengan cara yang sama. Misalkan ada 2 kromosom dengan panjang 12:

Induk 1: 0 1 1 1 0 | **0 1 0 1 1 0**

Induk 2: **1 1 0 1 0** | 0 0 0 1 1 0 1

Posisi yang dipilih: 5

Kromosom baru yang terbentuk:

Anak 1: 0 1 1 1 0 | 0 0 0 1 1 0 1

Anak 2: **1 1 0 1 0** | **0 1 0 1 1 0**

b. Crossover banyak titik

Proses *crossover* ini dilakukan dengan memisahkan suatu *string* menjadi beberapa bagian dan selanjutnya dipertukarkan dengan bagian dari *string* yang lain yang telah dipisahkan dengan cara yang sama sesuai dengan urutannya. Misalkan ada 2 kromosom dengan panjang 12:

Induk 1: **0 1** | **1 1 0 0** | **1 0 1 1** | **1 0**

Induk 2: 1 1 | 0 1 0 0 | 0 0 1 1 | 0 1

Posisi yang dipilih: 5

Kromosom baru yang terbentuk:

Anak 1: **0 1** | 0 1 0 0 | **1 0 1 1** | 0 1

Anak 2: 1 1 | **1 1 0 0** | 0 0 1 1 | **1 0**

c. Crossover seragam

Kromosom seragam menghasilkan kromosom keturunan dengan menyalin bit-bit secara acak dari kedua orang tuanya. Misalkan ada 2 kromosom dengan panjang 12.

Induk 1: **0 1 1 1 0 0 1 0 1 1 1 0**

Induk 2: **1 1 0 1 0 0 0 0 1 1 0 1**

Kromosom baru yang terbentuk:

Anak 1: **0 1 0 1 0 0 0 0 1 1 1 0**

Anak 2: **1 1 1 1 0 0 1 0 1 1 0 1**

d. Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Beberapa cara operasi mutasi diterapkan dalam algoritma genetika menurut jenis pengkodean terhadap *phenotype*, antara lain:

a. *Mutasi* dalam pengkodean biner

Mutasi pada pengkodean biner merupakan operasi yang sangat sederhana. Proses yang dilakukan adalah menginversi nilai bit pada posisi tertentu yang dipilih secara acak (atau dengan menggunakan skema tertentu) pada kromosom, yang disebut inversi bit.

Contoh mutasi pada pengkodean biner

Kromosom sebelum *mutasi*: 1 0 0 1 0 **1** 1 1

Kromosom sesudah *mutasi*: 1 0 0 1 0 **0** 1 1

b. *Mutasi* dalam pengkodean permutasi

Proses *mutasi* yang dilakukan dalam pengkodean biner dengan mengubah langsung bit-bit pada kromosom tidak dapat dilakukan pada pengkodean permutasi karena konsistensi urutan permutasi harus diperhatikan. Salah satu cara yang dapat dilakukan adalah dengan memilih dua posisi (*locus*) dari kromosom dan kemudian nilainya saling dipertukarkan.

Contoh *mutasi* dalam pengkodean permutasi

Kromosom sebelum *mutasi*: 1 2 **3** 4 6 5 8 **7** 9

Kromosom sesudah *mutasi*: 1 2 **7** 4 6 5 8 **3** 9

c. *Mutasi* dalam pengkodean nilai

Proses *mutasi* dalam pengkodean nilai dapat dilakukan dengan berbagai cara, salah satunya yaitu dengan memilih sembarang posisi gen pada kromosom, nilai yang ada tersebut kemudian ditambahkan atau dikurangkan dengan suatu nilai kecil tertentu yang diambil secara acak.

Contoh *mutasi* dalam pengkodean nilai riil dengan nilai yang ditambahkan atau dikurangkan adalah 0,1.

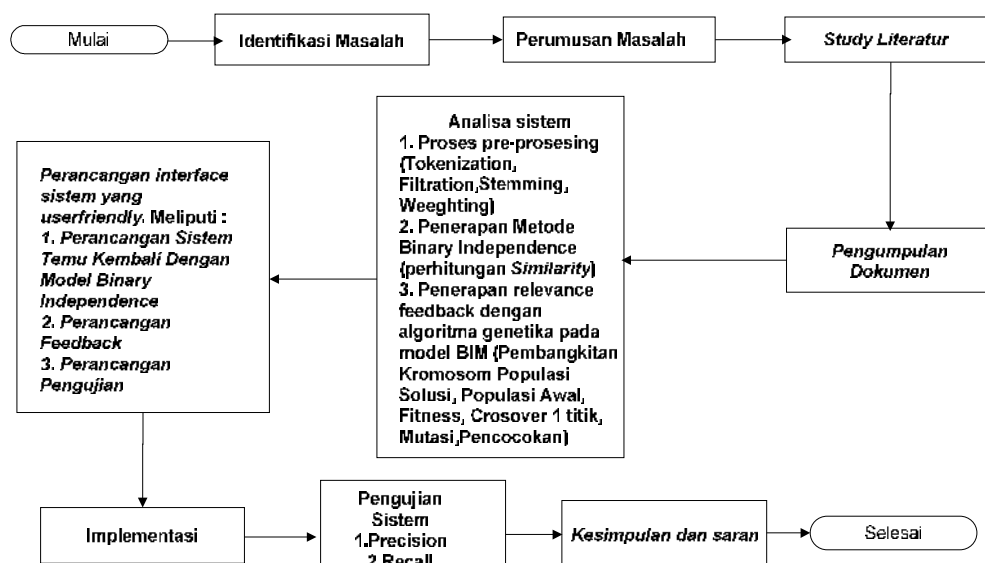
Kromosom sebelum *mutasi*: 1,43 **1,09** 4,51 **9,11** 6,94

Kromosom sesudah *mutasi*: 1,43 **1,19** 4,51 **9,01** 6,94

BAB III

METODOLOGI PENELITIAN

Metode penelitian merupakan sistematika yang menjelaskan tahapan penelitian yang akan dilaksanakan dalam pembuatan tugas akhir untuk dapat menjawab perumusan masalah penelitian. Tahapan yang akan dilakukan dalam penelitian ini adalah sebagai berikut:



Gambar 3.1 Tahapan Penyusunan Tugas Akhir

3.1 Identifikasi Masalah

Pada tahapan ini dilakukan tahapan indentifikasi permasalahan adalah bagaimana menyediakan informasi kepada *user* berdasarkan kebutuhan *user* melalui *query* yang di-*input*-kan oleh *user* tersebut dan menghasilkan *output* yang relevan terhadap *query*. Dengan menerapkan metode *relevance feedback* dalam sistem temu kembali informasi. Kemudian mengukur sejauh apa kerelevanan *output* yang dihasilkan oleh sistem dari sekumpulan dokumen yang ada.

3.2 Perumusan Masalah

Berdasarkan masalah yang telah diidentifikasi, maka dirumuskan masalah tentang bagaimana meningkatkan kinerja kerelevanan dokumen yang dihasilkan dari sistem temu kembali informasi berdasarkan *query* yang di-*input*-kan pengguna dan dapat menampilkan secara terangking berdasarkan tingkat kerelevanannya dari proses penerapan algoritma genetika pada *relevance feedback* dalam sistem temu kembali informasi dengan menggunakan *binary independence model*.

3.3 Study Literature

Tahapan ini adalah proses untuk mendapatkan informasi yang dibutuhkan untuk mendukung proses penelitian. Tahapan ini dilakukan dengan mengumpulkan jurnal-jurnal dan buku pendukung yang berkaitan dengan permasalahan pada penelitian tugas akhir. Sehingga diketahui model yang akan digunakan untuk dianalisa dan dibandingkan terhadap model lain dengan menggunakan metode tertentu.

3.4 Pengumpulan Dokumen

Tahapan ini adalah tahapan mengumpulkan koleksi dokumen (*corpus*), koleksi dokumen yang dijadikan penelitian adalah jurnal-jurnal teknologi yang dikumpulkan dari *download* di *internet*. Jumlah dokumen yang dijadikan bahan penelitian sebanyak 100 dokumen dimana dokumen-dokumen tersebut berformat *.txt*.

3.5 Analisa Sistem

Tahapan ini akan menjelaskan tentang proses dari membangun *relevance feedback* dalam sistem temu kembali informasi. Analisa sistem berarti metode yang khusus untuk menganalisa masalah dalam membangun sistem temu kembali informasi tersebut, sehingga mempermudah pemahaman terhadap sistem yang

akan dibangun dan dapat mencapai hasil akhir sesuai yang diinginkan. Analisa yang akan dilakukan adalah sebagai berikut:

1. Analisa dari tahapan pembangunan sistem temu kembali informasi, yaitu melakukan tahapan *pre-processing*. Tahapan ini adalah tahapan awal dalam membangun sistem temu kembali informasi sebelum menerapkan model-model dalam perhitungan nilai korelevanan (*similarity*), tahapan-tahapan tersebut adalah:
 - a. Pemisahan rangkaian kata (*tokenization*).
 - b. Penyaringan (*filtration*): penghapusan *stop-words*.
 - c. Konversi *term* ke bentuk kata dasar (*stemming*)
 - d. Pembobotan terhadap *term* (*weighting*).
2. Analisa berikutnya adalah analisa dalam menerapkan model perhitungan nilai korelevanan (*similarity*). Perhitungan itu dengan meneraokan *binary independence model* dalam penerapan sistem temu kembali informasi. Analisa ini digunakan untuk menggambarkan tahapan proses dalam *binary independence model*, bagaimana *binary independence model* dapat membandingkan korelevanan (*similarity*) terhadap *query* yang di-input-kan oleh *user* terhadap koleksi dokumen (*corpus*) yang terdapat dalam *database*. Sehingga pada akhirnya model *binary independence* dapat memberikan *output* dokumen yang relevan dari hasil pencarian yang dilakukan *user* berdasarkan *query* yang di-input-kan.
3. Analisa yang dilakukan setelah penerapan *binary independence model* adalah analisa *relevance feedback*. Analisa tersebut dengan melakukan penerapan algoritma genetika pada *relevance feedback* dalam sistem temu kembali informasi. Analisa ini untuk mengetahui bagaimana tahapan proses penerapan *relevance feedback*. Proses *feedback* dilakukan setelah

proses pencarian awal berdasarkan *query* pengguna selesai dilakukan. Proses *relevance feedback* dengan algoritma genetika yaitu:

- a. Pembangkitan Populasi Solusi
- b. Pembangkitan Populasi Awal
- c. Menghitung Fitness dengan Metode perhitungan Hsinchun Chen
- d. *Crossover* (*crossover* yang dilakukan dengan *crossover* satu titik)
- e. *Mutasi* (*mutasi* yang dilakukan dengan *mutasi* dengan pengodean biner)
- f. Pencocokan (yaitu pencocokan nilai kromosom dengan nilai *fitness* untuk menentukan dokumen yang relevan)

3.6 Perancangan Sistem

Tahapan perancangan sistem temu kembali informasi dengan menggunakan *relevance feedback* pada *binary independence model* berdasarkan analisa bertujuan untuk memberikan kemudahan dan menyederhanakan suatu proses, perancangan terhadap model, dan merancang bangun sistem untuk dapat dimengerti oleh *user*. Rancangan utama yang akan dilakukan adalah merancang *interface* yang *user friendly* agar mudah digunakan dan dipahami oleh *user*. Perancangan sistem yang akan dibangun berdasarkan tiga tahapan yang ingin dicapai, yaitu:

- a. Perancangan *interface* sistem temu kembali informasi dengan *binary independence model*
- b. Perancangan *interface* sistem *relevance feedback*
- c. Perancangan *interface* sistem pengujian

3.7 Implementasi

Tahapan implementasi akan dilakukan dengan pembuatan modul-modul yang telah dirancang dan dianalisa, selanjutnya diimplementasikan dalam perancangan kedalam bahasa pemrograman. Implementasi sistem akan dilakukan dengan spesifikasi sebagai berikut:

a. Perangkat Keras

Processor : *Pentium® dual-core T2390, 1,86GHz*

Memori (RAM) : 2 GB

b. Perangkat Lunak

Sistem Operasi : *Microsoft Windows XP3.*

Bahasa Pemrograman : PHP

Web Browser : *Firefox*

DBMS : MySQL

3.8 Pengujian Sistem

Pentingnya pengujian perangkat lunak dan implikasinya yang mengacu pada kualitas perangkat lunak tersebut. Maka dalam bahasan pengujian sistem ini adalah tahapan menjalankan sistem temu kembali informasi, guna mengetahui apakah sistem yang dibangun telah beroperasi sesuai target yang diinginkan. Dimana dalam hal ini apakah sistem temu kembali informasi dapat memberikan *output* yang relevan sesuai dengan *query* yang diinputkan oleh *user*. Ukuran dari keberhasilan sistem temu kembali informasi adalah melihat perbandingan dari dokumen yang dihasilkan oleh sistem temu kembali informasi.

Penilaian pengujian dilakukan terhadap beberapa aspek untuk memastikan dari kualitas pengujian sehingga dapat memberikan hasil kesimpulan dan saran yang baik untuk pengembangan sistem kedepannya. Penilaian tersebut meliputi:

- a. Pengujian *precision* dan *recall* dari *output* yang dihasilkan oleh sistem, yaitu sistem temu kembali informasi dengan *binary independence model*

- b. Pengujian *precision* dan *recall* dari *output* yang dihasilkan dari penerapan metode *relevance feedback* dengan algoritma genetika.
- c. Pengujian *precision* dan *recall* dari *output* yang dipengaruhi oleh panjang atau pendeknya *query* yang di *input* oleh *user*.
- d. Pengujian *precision* dan *recall* dari *output* yang dipengaruhi oleh relevan atau tidaknya dokumen yang dijadikan *feedback* oleh *user*.

Pengujian *precision* adalah pengujian terhadap perbandingan dokumen relevan yang ditemukan terhadap semua dokumen yang berhasil ditemukan. Dan pengujian *recall* adalah pengujian terhadap perbandingan dokumen relevan yang ditemukan terhadap semua dokumen relevan yang ada.

3.9 Kesimpulan dan Saran

Tahapan kesimpulan dan saran adalah tahapan akhir pada penelitian tugas akhir ini. Tahapan ini akan membahas tentang kesimpulan yang dihasilkan dari penelitian tugas akhir, kesimpulan diambil dari proses analisa sampai kepada implementasi dan pengujian.

Pada tahapan saran, tugas akhir ini diharapkan dapat memberikan sebuah catatan rekomendasi untuk menyempurnakan dan mengembangkan penelitian sistem temu kembali informasi selanjutnya, terutama yang berkaitan dengan penerapan *relevance feedback* dalam *binary independence model*.

BAB IV

ANALISA DAN PERANCANGAN

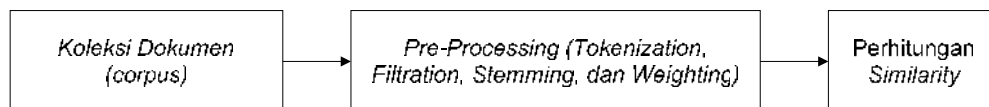
Bab ini menjelaskan tentang proses dari membangun sistem temu kembali informasi. Analisa sistem berarti metode yang khusus untuk menganalisa masalah dalam membangun sistem temu kembali informasi. Tahapan ini akan menganalisa alur proses penerapan *Binary Independence* dalam sistem temu kembali dan proses penerapan algoritma genetika pada *relevance feedback* dalam sistem temu kembali informasi yang menggunakan *binary independence model*.

Tahapan selanjutnya, adalah tahapan perancangan dimana bertujuan mempermudah dan menyederhanakan suatu proses atau jalannya aliran data, perancangan terhadap model, dan merancang bangun sistem untuk dapat dimengerti *user*. Perancangan terhadap model, perancang sistem temu kembali informasi sesuai dengan penerapan *binary independence model* dan perancangan *relevance feedback* dalam sistem temu kembali informasi yang menggunakan algoritma genetika.

4.1 Analisa Sistem Temu Kembali Informasi

Secara garis besar tahapan arsitektur sistem temu kembali informasi ada tiga proses yang dilakukan, yaitu mengumpulkan koleksi dokumen, melakukan tahapan *preprocessing* terhadap koleksi dokumen, dan tahapan berikutnya adalah menerapkan metode tertentu untuk menghitung kedekatan (relevansi atau *similarity*). Dalam kasus ini perhitungan *similarity* menggunakan *binary independence model*.

Proses dari tahapan sistem temu kembali informasi dapat dilihat dari Gambar 4.1, tahapannya sebagai berikut:



Gambar 4.1 Alur Peroses Sistem Temu Kembali Informasi

4.1.1. Pengumpulan Dokumen

Tahapan ini adalah tahapan mengumpulkan koleksi dokumen (*corpus*), koleksi dokumen yang dijadikan penelitian adalah jurnal-jurnal teknologi yang dikumpulkan dari *download* di *internet*. Koleksi dokumen (*corpus*) tersebut berbahasa indonesia. Jumlah dokumen yang dijadikan bahan penelitian sebanyak 100 dokumen dimana dokumen-dokumen tersebut berformat *txt*. Koleksi dokumen tersebut di simpan didalam *dbms* MySQL.

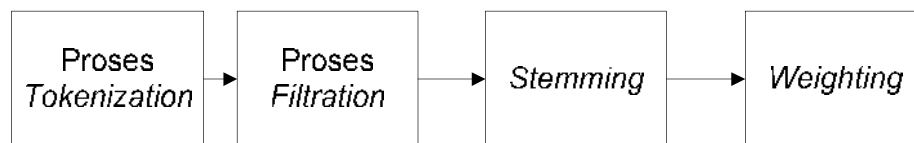
Misalkan dalam *database corpus* terdapat 14 dokumen yang menjadi target pencarian. Koleksi dokumen tersebut adalah sebagai berikut:

- d 1 : Pemerintah hanya menjual selang dan regulator tabung elpiji, melalui agen resmi.
- d 2 : Agen resmi menggelapkan selang-selang dan beberapa regulator?
- d 3 : “Tabung elpiji ditarik oleh Pemerintah”.
- d 4 : Tabung elpiji milik Pemerintah, berbeda dengan tabung elpiji milik agen resmi.
- d 5 : Selang dan regulator tabung elpiji SNI, berbeda dengan selang dan regulator saat ini.
- d 6 : Sistem informasi sangat penting bagi lembaga pendidikan.
- d 7 : Semua orang membutuhkan informasi.
- d 8 : Sistem informasi sekolah berbasis web dan desktop.
- d 9 : Hasil kajian tim UI dijadikan acuan untuk penyusunan cetak biru pemanfaatan teknologi informasi dan komunikasi (ICT) di sektor pendidikan.
- d10 : Indonesia tergolong bangsa digital divide, yaitu kesenjangan penguasaan, pemahaman serta pemanfaatan teknologi informasi oleh masyarakatnya.
- d11 : Telah disadari dan merupakan konsensus bahwa teknologi informasi dan komunikasi (ICT) adalah pusat dari penciptaan ekonomi global berbasis ilmu pengetahuan.
- d12 : Perkembangan teknologi Informasi yang sedemikian pesat telah memicu munculnya berbagai teknologi informasi baru untuk memenuhi kebutuhan manusia.

- d13 : Film ini diawali dengan cerita tentang 'Palestina tahun 1948' yang menggambarkan perjalanan para pengungsi Arab.
- d14 : Bukti tidak adanya koordinasi antara otoritas moneter dengan otoritas fiskal adalah apa yang terjadi pada akhir bulan Agustus 2001 lalu.

4.1.2. *Preprocessing*

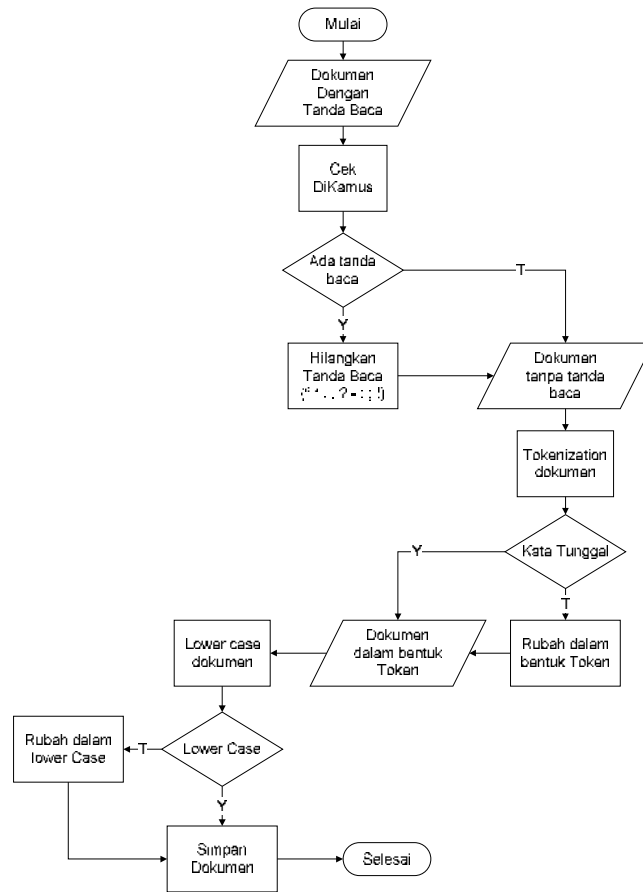
Pada tahapan *preprocessing* ada beberapa proses yang akan dilakukan oleh sistem, pada gambar 4.2 berikut adalah bagan dari tahapan proses dalam *preprocessing*:



Gambar 4.2 Alur Peroses *Preprocessing*

4.1.2.1 . *Tokenization*

Pada tahapan *preprocessing* yang pertama adalah tahapan *tokenization*. Koleksi dokumen diatas akan dilakukan tahapan *preprocessing* yaitu tahapan *tokenization*. Tahapan *tokenization* akan menghilangkan seluruh tanda baca, mengubah kalimat menjadi bentuk *token* dan mengubah huruf besar menjadi huruf kecil (*lower case*) yang terdapat dalam koleksi dokumen. Jika digambarkan ke dalam bentuk *flowchart* maka proses *tokenization* digambarkan pada Gambar 4.2, yaitu sebagai berikut:



Gambar 4.3 Flowchart Tokenization

Hasil dari proses *tokenization* dari koleksi dokumen diatas adalah:

Tabel 4.1 Hasil Proses *Tokenization*

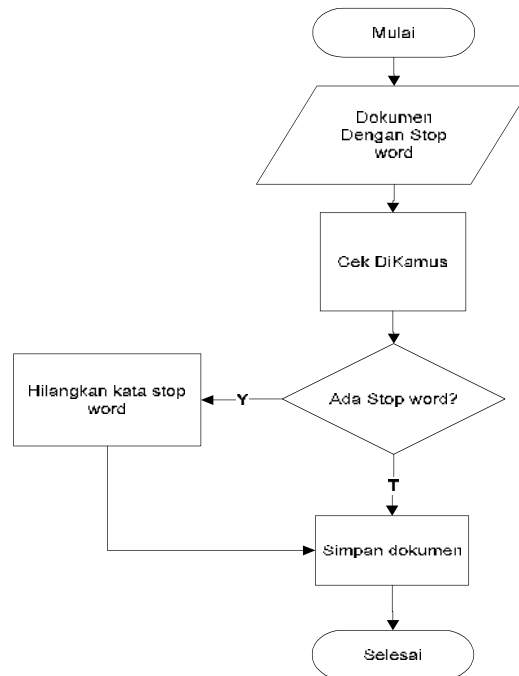
dok 1	dok 2	dok 3	dok 4	dok 5
pemerintah	agen	tabung	tabung	selang
hanya	resmi	elpiji	elpiji	dan
menjual	menggelapkan	ditarik	milik	regulator
selang	selang	oleh	pemerintah	tabung
dan	selang	pemerintah	berbeda	elpiji
regulator	dan		dengan	sni
tabung	beberapa		tabung	berbeda
elpiji	regulator		elpiji	dengan
melalui			milik	selang
agen			agen	dan
resmi			resmi	regulator
				saat
				ini

Tabel 4. 1 Lanjutan Hasil Proses *Tokenization*

dok 6	dok 7	dok 8	dok 9	dok 10
system	semua	sistem	hasil	indonesia
informasi	orang	informasi	kajian	tergolong
sangat	membutuhkan	sekolah	tim	bangsa
penting	informasi	berbasis	ui	digital
bagi		web	dijadikan	divide
lembaga		dan	acuan	yaitu
pendidikan		desktop	untuk	kesenjangan
			penyusunan	penguasaan
			cetak	pemahaman
			biru	serta
			pemanfaatan	pemanfaatan
			teknologi	teknologi
			informasi	informasi
			dan	oleh
			komunikasi	masyarakat
			ict	
			disektor	
			pendidikan	
dok 11	dok 12	dok 13	dok 14	
telah	perkembangan	film	bukti	
disadari	teknologi	ini	tidak	
dan	informasi	diawali	adanya	
merupakan	yang	dengan	koordinasi	
konsensus	sedemikian	cerita	antara	
bahwa	pesat	tentang	otoritas	
teknologi	telah	palestina	moneter	
informasi	memicu	tahun	dengan	
dan	munculnya	1948	otoritas	
komunikasi	berbagai	yang	fisikal	
ict	teknologi	menggambarkan	adalah	
adalah	informasi	perjalanan	apa	
pusat	baru	para	yang	
dari	untuk	pengungsi	terjadi	
penciptaan	memenuhi	arab	pada	
ekonomi	kebutuhan		akhir	
global	manusia		bulan	
berbasis			agustus	
ilmu			2001	
pengetahuan			lalu	

4.1.2.2. Filtration

Tahapan selanjutnya setelah proses *tokenization* adalah proses *filtration*. Dalam proses ini dokumen akan di seleksi *term-term* yang termasuk kedalam *stopword* dan akan dihapus dari koleksi dokumen. Proses *filtration* jika digambarkan dalam bentuk *flowchart*, maka sebagai berikut:



Gambar 4.4 Flowchart Stopword

Hasil dari proses *filtration* dari proses diatas adalah sebagai berikut:

Tabel 4.2 Hasil Proses Filtration

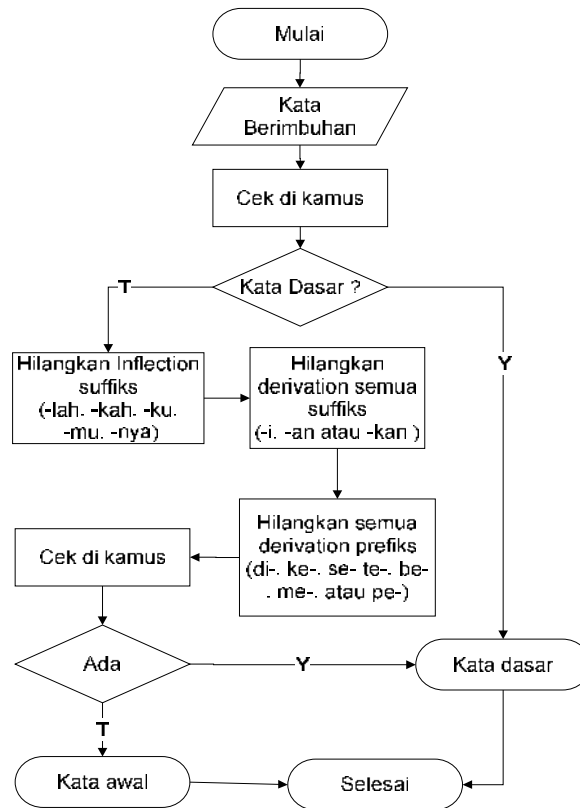
dok 1	dok 2	dok 3	dok 4	dok 5
pemerintah	agen	tabung	tabung	selang
menjual	resmi	elpiji	elpiji	regulator
selang	menggelapkan	ditarik	milik	tabung
regulator	selang	pemerintah	pemerintah	elpiji
tabung	selang		berbeda	sni
elpiji	beberapa		dengan	berbeda
melalui	regulator		tabung	selang
agen			elpiji	regulator
resmi			milik	
			agen	
			resmi	

Tabel 4.2 Lanjutan Hasil Proses *Filtration*

dok 6	dok 7	dok 8	dok 9	dok 10
system	orang	sistem	hasil	indonesia
informasi	membutuhkan	informasi	kajian	tergolong
sangat	informasi	sekolah	tim	bangsa
penting		berbasis	ui	digital
lembaga		web	dijadikan	divide
pendidikan		desktop	acuan	kesenjangan
			penyusunan	penguasaan
			cetak	pemahaman
			biru	pemanfaatan
			pemanfaatan	teknologi
			teknologi	informasi
			informasi	masyarakat
			komunikasi	
			ict	
			disektor	
			pendidikan	
dok 11	dok 12	dok 13	dok 14	
disadari	perkembangan	film	bukti	
merupakan	teknologi	diawali	koordinasi	
konsensus	informasi	cerita	otoritas	
teknologi	pesat	tentang	moneter	
informasi	memicu	palestina	otoritas	
komunikasi	munculnya	tahun	fisikal	
ict	berbagai	1948	akhir	
pusat	teknologi	menggambarkan	bulan	
penciptaan	informasi	para	agustus	
ekonomi	baru	pengungsi	2001	
global	memenuhi	arab		
berbasis	kebutuhan			
ilmu	manusia			
pengetahuan				

4.1.2.3. *Stemming*

Setelah tahapan *filtration*, maka tahapan berikutnya dari tahapan *preprocessing* adalah tahapan *stemming*. Tahapan *stemming* yang di terapkan adalah *stemming* algoritma Nazief dan Andriani. Tahapan yang dilakukan pada *stemming* dapat dijelaskan pada Gambar 4.5 yaitu *flowchart* berikut:



Gambar 4.5 Flowchart Algoritma Nazief dan Adriani

Hasil proses *stemming* dari proses diatas adalah sebagai berikut:

Tabel 4.3 Hasil Proses *Stemming*

dok 1	dok 2	dok 3	dok 4	dok 5
pemerintah	agen	tabung	tabung	selang
jual	resmi	elpiji	elpiji	regulator
selang	gelap	tarik	milik	tabung
regulator	selang	pemerintah	pemerintah	elpiji
tabung	selang		beda	sni
elpiji	regulator		tabung	beda
agen			elpiji	selang
resmi			milik	regulator
			agen	
			resmi	

Tabel 4.3 Lanjutan Hasil Proses *Stemming*

dok 6	dok 7	dok 8	dok 9	dok 10
sistem	orang	sistem	hasil	indonesia
informasi	butuh	informasi	kajian	golong
penting	informasi	sekolah	tim	bangsa
lembaga		bsis	ui	digital
didik		web	acuan	divide
		desktop	susun	senjang
			cetak	kuasa
			biru	paham
			manfaat	manfaat
			teknologi	teknologi
			informasi	informasi
			komunikasi	masyarakat
			ict	
			sektor	
			didik	
dok 11	dok 12	dok 13	dok 14	
sadar	kembang	film	bukti	
konsensus	teknologi	awal	koordinasi	
teknologi	informasi	cerita	otoritas	
informasi	pesat	palestina	moneter	
komunikasi	picu	tahun	otoritas	
ict	muncul	1948	fisikal	
pusat	teknologi	gambar	akhir	
cipta	informasi	ungsi	bulan	
ekonomi	baru	arab	agustus	
global	butuh		2001	
basis	manusia			
ilmu				
pengetahuan				

4.1.2.4 Pembobotan Kata (*Weighting*).

Tahapan selanjutnya setelah tahapan *stemming* dilakukan adalah tahapan pembobotan terhadap kata (*term*) yang terdapat dalam koleksi dokumen. Pembobotan dilakukan dengan memberikan nilai *biner* 1 atau 0 terhadap ada atau tidak *term* pada koleksi dokumen. Pembobotan kata tersebut berdasarkan model *binary independence* yaitu dengan menghitung nilai dokumen frekuensi dan nilai

sebuah dokumen yang tidak relevan mengandung kata pada dokumen. Maka hasil *weighting* dari model *binary independence* adalah sebagai berikut:

Tabel 4.4 Hasil Proses *weighting*

Term	d	d	d	d	d	d	d	d	d	d1	D	D	D	D	df	Qk (n/N)
	1	2	3	4	5	6	7	8	9		1	1	1	1		
pemerintah	1	0	1	1	0	0	0	0	0	0	0	0	0	0	3	0.23077
jual	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0.07692
selang	1	2	0	0	2	0	0	0	0	0	0	0	0	0	3	0.23077
regulator	1	1	0	0	2	0	0	0	0	0	0	0	0	0	3	0.23077
tabung	1	1	1	2	1	0	0	0	0	0	0	0	0	0	4	0.30769
elpiji	1	0	1	2	1	0	0	0	0	0	0	0	0	0	4	0.30769
agen	1	1	0	1	0	0	0	0	0	0	0	0	0	0	3	0.23077
resmi	1	1	0	1	0	0	0	0	0	0	0	0	0	0	3	0.23077
gelap	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0.07692
tarik	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0.07692
milik	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1	0.07692
sni	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0.07692
system	0	0	0	0	0	1	0	1	0	0	0	0	0	0	2	0.15385
informasi	0	0	0	0	0	1	1	1	1	1	1	2	0	0	7	0.53846
sekolah	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0.07692
penting	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0.07692
lembaga	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0.07692
didik	0	0	0	0	0	1	0	0	1	0	0	0	0	0	2	0.15385
orang	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0.07692
butuh	0	0	0	0	0	0	1	0	0	0	0	1	0	0	2	0.15385
basis	0	0	0	0	0	0	0	1	0	0	1	0	0	0	2	0.15385
web	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0.07692
destop	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0.07692
hasil	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
kajian	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
tim	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
ui	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
acuan	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
susun	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
cetak	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
biru	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
manfaat	0	0	0	0	0	0	0	0	1	1	0	0	0	0	2	0.15385
teknologi	0	0	0	0	0	0	0	0	1	1	1	2	0	0	4	0.30769

Tabel 4.4 Lanjutan Hasil Proses *Weighting*

Term	d 1	d 2	d 3	d 4	d 5	d 6	d 7	d 8	d 9	d1 0	D 1 1	D 1 2	D 1 3	D 1 4	df	Qk (n/N)
ict	0	0	0	0	0	0	0	0	1	0	1	0	0	0	2	0.15385
sector	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0.07692
indonesia	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
golong	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
bangsa	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
digital	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
divide	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
senjang	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
kuasa	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
paham	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
masyarakat	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0.07692
sadar	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
konsensus	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
pusat	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
cipta	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
ekonomi	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
global	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
ilmu	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
pengetahuan	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0.07692
kembang	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
pesat	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
picu	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
muncul	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
baru	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
penuh	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
manusia	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.07692
film	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
cerita	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
palestina	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
tahun	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
1948	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
gambar	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
jalan	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
ungsi	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
arab	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0.07692
bukti	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692
koordinasi	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692

Tabel 4.4 Lanjutan Hasil Proses *Weighting*

Term	d 1	d 2	d 3	d 4	d 5	d 6	d 7	d 8	d 9	d1 0	D 1 1	D 1 2	D 1 3	D 1 4	df	Qk (n/N)
moneter	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692
fiskal	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692
akhir	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692
bulan	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692
agustus	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692
2001	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0.07692

4.1.3. Perhitungan Nilai *Similarity Binary Independence Model*

Setelah tahapan *preprocessing* selesai dilakukan, maka tahapan dalam sistem temu kembali informasi adalah menerapkan model perhitungan korelevanan dokumen dari hasil pembobotan. Dari tabel 4.4, maka perhitungan nilai *similarity* tiap dokumen dengan menggunakan perhitungan model *binary independence* pada rumus persamaan 2.7 adalah sebagai berikut:

Nilai dari $P_k(1-q_k)$ dan $q_k(1-P_k)$ untuk masing-masing *query*:

Informasi : $P_k(1-q_k) : 0.23077$; $Q_k(1-P_k) : 0.269231$

Otoritas : $P_k(1-q_k) : 0.46154$; $Q_k(1-P_k) : 0.038462$

Pemerintah : $P_k(1-q_k) : 0.38462$; $Q_k(1-P_k) : 0.115385$

Perhitungan *similarity*:

$$\begin{aligned} \text{Dok 1} &= \left| \sum 0. \log \frac{1}{1} + 0. \log \frac{1}{1} + 1. \log \frac{1}{1} \right| \\ &= \left| 0 + 0 + 0.522879 \right| = 0.522879 \end{aligned}$$

$$\begin{aligned} \text{Dok 2} &= \left| 0. \log \frac{1}{1} + 0. \log \frac{1}{1} + 0. \log \frac{1}{1} \right| \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Dok 3} &= \left| \sum 0. \log \frac{1}{1} + 0. \log \frac{1}{1} + 1. \log \frac{1}{1} \right| \\ &= \left| 0 + 0 + 0.522879 \right| = 0.522879 \end{aligned}$$

$$\begin{aligned} \text{Dok 4} &= \left| \sum 0. \log \frac{1}{1} + 0. \log \frac{1}{1} + 1. \log \frac{1}{1} \right| \\ &= \left| 0 + 0 + 0.522879 \right| = 0.522879 \end{aligned}$$

$$\begin{aligned} \text{Dok 5} &= \left| 0. \log \frac{1}{1} + 0. \log \frac{1}{1} + 0. \log \frac{1}{1} \right| \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Dok 6} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 7} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 8} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 9} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 10} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 11} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 12} &= \left| 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= \left| -0.06694 + 0 + 0 \right| = 0.06694 \end{aligned}$$

$$\begin{aligned} \text{Dok 13} &= \left| 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Dok 14} &= \left| 0. \text{Log} \frac{1}{2} + 1. \text{Log} \frac{1}{2} + 0. \text{Log} \frac{1}{2} \right| \\ &= 0 + 0.52287 + 0 = 0.52287 \end{aligned}$$

Dari perhitungan diatas, maka dokumen yang ditemukan oleh perhitungan *binary independence model* adalah: dok1,dok3,dok,5,dok,14,dok6,dok7,dok8,dok 9,dok10,dok11,dok12. Sedangkan dokumen yang tidak ditemukan adalah : dok2, dok5, dok 13.

4.2 Analisa *Relevance Feedback* Dengan Algoritma Genetika.

Proses perhitungan kemiripan dokumen dalam *relevance feedback*, antara dokumen yang dijadikan umpan balik (*feedback*) dengan koleksi dokumen (*corpus*) yaitu dengan menggunakan algoritma genetika. Penerapan algoritma genetika dimulai dari proses sebagai berikut:

4.2.1 Penentuan populasi solusi

Dokumen yang di *feedback* adalah dokumen yang dipilih oleh pengguna dalam pencarian dengan *binary independence model* yaitu misalkan: Dokumen 1 *dl*: Pemerintah hanya menjual selang dan regulator tabung elpiji, melalui agen resmi.

4.2.2. Penentuan populasi awal

Dokumen yang menjadi *feedback* di jadikan sebagai populasi solusi
pemerintah + jual + selang + regulator + tabung + elpiji + agen + resmi

4.2.3. Mengubah Populasi Solusi Kedalam Kromosom

Proses membangkitkan kromosom polulasi solusi yaitu dengan cara memberikan bobot biner bernilai 1 terhadap semua *term* pada dokumen *feedback*.

Tabel 4.5 Pembangkitan Kromosom Populasi Solusi

pemerintah	jual	selang	regulator	tabung	elpiji	agen	resmi
1	1	1	1	1	1	1	1

Sehingga kromosomnya : 1111 1111 = ada 8 kromosom

4.2.4. Pembangkitan Populasi Awal Kedalam Kromosom Populasi Solusi

Pada Tabel 4.6 berikut adalah hasil pembangkitan populasi awal kedalam populasi solusi.

Tabel 4.6 Pembangkitan Kromosom Populasi Awal.

Doc	Term							
	pemerintah	jual	selang	regulator	tabung	elpiji	agen	resmi
1	0	0	1	1	1	0	1	1
2	1	0	0	0	1	1	0	0
3	1	0	0	0	1	1	1	1
4	0	0	1	1	1	1	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0		0	0	0	0	0
8	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0

Proses pembangkitan populasi awal kedalam bentuk kromosom populasi solusi pada tabel 4,6 di atas dilakukan dengan cara memberikan bobot biner bernilai 1 atau 0,sesuai dengan ada atau tidaknya *term* pada koleksi dokumen terhadap *term* yang menjadi populasi solusi

Pengkodean kromosom populasi solusi dilakukan terhadap kromosom yang memiliki gen sesuai populasi awal dengan populasi solusi. Hasil pengkodean kromosom populasi:

Tabel 4.7 Pengkodean Kromosom Populasi Solusi

No Populasi	No Doc	Kromosom
1	2	0011 1011
2	3	1000 1100
3	4	1000 1111
4	5	0011 1100

4.2.5 Hitung Nilai *Fitness*

Untuk menghitung nilai *fitness*, digunakan persamaan rumus 2.8 berikut adalah tabel hasil perhitungan nilai *fitness*.

Tabel 4.8 Hasil *Fitness*

No Doc	Kromosom	di (jk)	di(j)	Nilai Fitness
2	0011 1011	5	8	0.625
3	1000 1100	3	8	0.375
4	1000 1111	5	8	0.625
5	0011 1100	4	8	0.5

Contoh cara perhitungan *fitness* pada tabel 4.8 digunakan persamaan rumus 2.8 adalah sebagai berikut:

$$Fitness \text{ no dokumen } 2 = \frac{(5)}{(8)} = 0.625$$

4.2.6 *Crossover*

Di ambil 2 nilai *fitness* tertinggi untuk di *crossover*

Doc 2 : 0.625 dan dok 4 : 0.625

Crossover dilakukan dengan metode Satu titik, pemilihan *crossover* satu titik karena untuk mempermudah sistem temu kembali informasi dalam memberikan hasil dokumen yang di *feedback* oleh pengguna dengan cepat jika dibandingkan dengan *crossover* banyak titik atau *crossover* seragam. Hal ini dikarenakan sistem tidak perlu membuat segmentasi pemisah gen menjadi beberapa bagian, sistem cukup memisahkan gen menjadi dua bagian, yaitu dengan menentukan titik tengah kromosom atau median dari kromosom.

Doc 2 = **0011** 1011 menjadi **0011 1111** nilai fitness 0.75

Doc 4 = 1000 **1111** menjadi 1000 1011 nilai fitness 0.5

4.2.7 *Mutasi*

Mutasi dilakukan dengan metode pengkodean biner, yaitu mengubah titik tertentu dari kromosom yang bernilai 0 menjadi bernilai 1 atau sebaliknya. Pemilihan mutasi dengan metode pengkodean biner akan mempermudah sistem melakukan mutasi karena pada model *binary independence* kemunculan tiap *term* pada dokumen di bobot dengan nilai biner. Dalam hal ini titik ditentukan pada nilai *median*. Nilai kromosom yang dipilih adalah kromosom yang memiliki nilai *fitness* terbesar dari hasil *crossover*.

Yaitu:

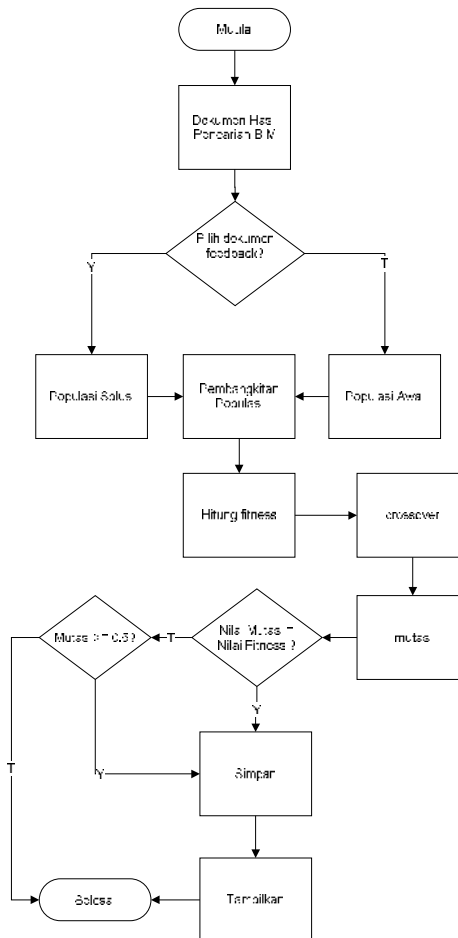
Hasil *crossover* dengan nilai tertinggi 0011 1111 = 0.75

di *mutasi* menjadi 0010 1111 = 0.625

4.2.8 Pencocokan dokumen

Nilai mutasi dijadikan untuk mencari dokumen mana saja yang bernilai *fitness* sama dengan nilai mutasi, maka dokumen itu yang dianggap paling relevan, yaitu dokumen dokumen 2 dan dokumen 4. Maka yang akan ditampilkan oleh sistem kepada pengguna adalah dokumen 2, dokumen 4.

Alur proses *relevance feedback* dengan algoritma genetika dalam sistem temu kembali informasi pada tahapan-tahapan diatas dapat digambarkan sebagai berikut:



Gambar 4. 6 Alur Proses *Relevace Feedback* dengan Algoritma Genetika

4.3 Perancangan Antarmuka (*Interface*) Sistem

Tahapan ini dilakukan dengan tujuan untuk dapat merancang antarmuka sistem yang akan dibangun dengan sebaik-baiknya sehingga sistem dapat menjadi *user friendly* bagi para penggunanya. Adapun beberapa dari rancangan tampilan tersebut, yaitu :

1. Tampilan Menu Utama (Menu Pencarian)

Gambar 4.7 adalah menu utama dalam sistem temu kembali, menu ini akan tampil ketika sistem dijalankan. Menu utama pada Gambar 4.7 yaitu proses *retrieval* (pencarian) yang dapat di akses oleh pengguna. Pengguna dapat langsung meng-*input*-kan *query* dan melakukan pencarian.

Gambar 4.7 Rancangan Menu Utama (Menu Pencarian)

2. Hasil Pencarian Menu Utama

Gambar 4.8 akan menampilkan hasil pencarian dari *query* yang pengguna masukan dalam menu pencarian pada menu utama sistem temu kembali informasi. Pada menu hasil pencarian akan memberikan hasil pencarian secara terangkan. Dokumen yang di tampilkan adalah terangkan berdasarkan nilai *similarity* pada perhitungan model *binary independence* yang dilakukan sistem. Dokumen yang muncul adalah sebagian dari isi dokumen yang menggambarkan hasil pencarian.

Gambar 4.8 Rancangan Hasil Pencarian Menu Utama

3. Tampilan hasil pencarian dari *feedback*

Tampilan Gambar 4.9 akan memberikan hasil pencarian dari proses *feedback* yang dilakukan oleh pengguna pada hasil pencarian pertama. Dokumen hasil pencarian akan ditampilkan secara terrangking berdasarkan hasil perhitungan pada algoritma genetikan. Perbedaan *form* hasil *feedback* pengguna biasa dengan pengguna admin pada informasi detail perhitungan dokumen tersebut.

Gambar 4.9 Rancangan Hasil Pencarian *Feedback*

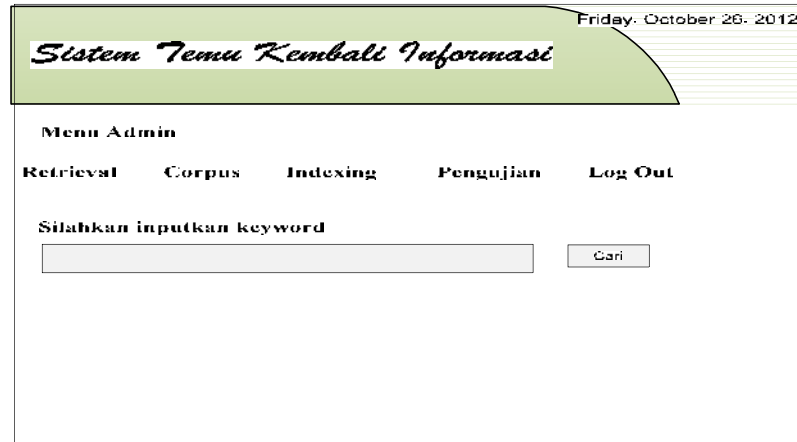
4. Tampilan Menu *Login*

Menu *login* dalam sistem berfungsi untuk membedakan hak akses dalam menggunakan sistem. *Login* akan membedakan *user* sebagai *admin* atau *user* sebagai pengguna pengunjung sistem. Gambar 4.10 adalah tampilan ketika pengguna akan mengakses dan mengelola sistem sebagai *admin*. Terdapat menu *login* sebagai *admin*, yaitu dengan input *username* dan *password*.

Gambar 4.10 Rancangan Menu *Login*

5. Tampilan menu pencarian *admin*

Gambar 4.11 merupakan rancangan dari menu pencarian sistem temu kembali informasi pada admin. Menu ini sama seperti pada menu utama. *Admin* menginputkan *query* kemudian melakukan pencarian dan sistem akan menampilkan dokumen yang relevan sesuai dengan *query admin* tersebut.



Friday, October 26, 2012

Sistem Temu Kembali Informasi

Menu Admin

Retrieval Corpus Indexing Pengujian Log Out

Silahkan inputkan keyword

Gambar 4.11 Rancangan Menu Pencarian *Admin*

6. Tampilan Hasil Pencarian

Tampilan Gambar 4.12 merupakan *form* yang hampir sama dengan *from* pada menu hasil pencarian utama, dimana *form* akan menampilkan hasil pencarian pertama dari proses *binary independence model*. *Form* akan memberikan informasi hasil pencarian berdasarkan *query* yang pengguna *input* pada menu sebelumnya.

Tampilan Gambar 4.12 juga akan memberikan pilihan dokumen mana yang akan di pilih oleh *user* untuk dijadikan *feedback* jika *user* merasa dokumen yang *user* cari belum *relevance* atau ingin mencari kemiripan dokumen lainnya. Perbedaan tampilan ini adalah dari segi hak aksesnya kemudian tampilan ini memberikan menu penilaian apakah dokemen yang ditemukan dari hasil pencarian relevan atau tidak. Hal ini berfungsi untuk pengujian sistem.

Friday, October 26, 2012

Sistem Temu Kembali Informasi

Retrieval Corpus Indexing Pengujian Log out

Retrieval

Silahkan inputkan keyword

Hasil Query

Txi Dokumen Hasil Pencarian

Hasil 1

Hasil 2

Gambar 4.12 Rancangan Hasil Pencarian BIM

7. Tampilan *Upload Corpus*

Gambar 4.13 adalah tampilan menu untuk *upload copus*, dimana menu ini hanya dapat diakses oleh admin jika ingin menambah *corpus*. Admin mengambil dimana lokasi dokumen yang ingin di *upload*, kemudian melakukan *upload*. Kemudian sistem akan melakukan proses *preprocessing* secara otomatis.

Friday, October 26, 2012

Sistem Temu Kembali Informasi

Menu Admin

Retrieval Corpus Indexing Pengujian Log Out

Upload Corpus → List Corpus

CORPUS

Silakan upload file CORPES

Invalid file

Gambar 4.13 Rancangan Menu *Upload Corpus*

8. Tampilan *Indexing*

Gambar 4.14 adalah tampilan hasil dari proses *indexing*. Menu akan memberikan informasi tentang keberhasilan *preprocessing* dan menu untuk *indexing corpus*. Menu ini hanya dapat diakses oleh *admin*.

Friday, October 26, 2012

Sistem Temu Kembali Informasi

Menu Admin

Retrieval Corpus Indexing Pengujian Log Out

Indexing

Total Dokumen =
 Total Kata =
 Status saat Yang Diindex =
 Klik Link Berikut untuk index ulang

List Dokumen		View Detail
No	Nama File	Jumlah Kata

Page 1

Gambar 4.14 Rancangan Menu Proses Hasil *Indexing*

9. Tampilan Detail *Indexing*

Gambar 4.15 adalah menu yang akan memberikan informasi daftar *corpus* yang tersimpan dalam *database* yang telah dilakukan *indexing* secara detail. Proses ini hanya dapat diakses oleh *admin*.

Tuesday, October 18, 2012

Sistem Temu Kembali Informasi

Menu Admin

Retrieval Corpus Indexing Pengujian Log Out

Indexing

Total Dokumen =
 Total Kata =
 Status saat Yang Diindex =
 Klik Link Berikut untuk index ulang

Id	Term	Jumlah Dokumen	Id_dok	df	qk	pk	qkpk	logukpk

Page 1

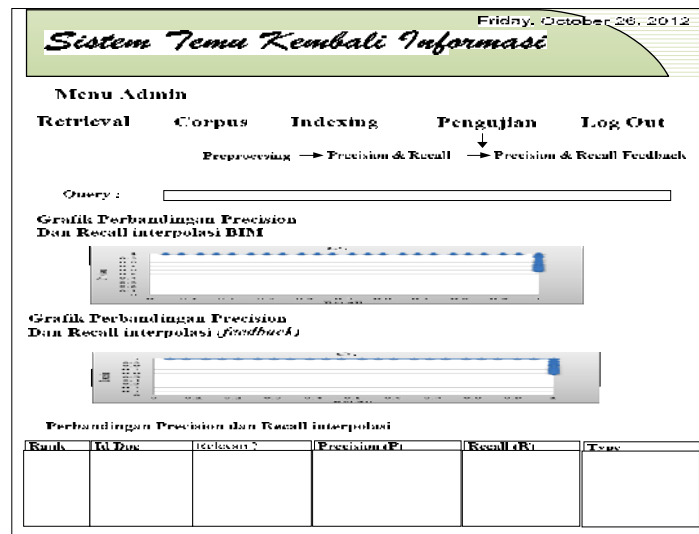
List Dokumen		View Detail
No	Nama File	Jumlah Kata

Page 1

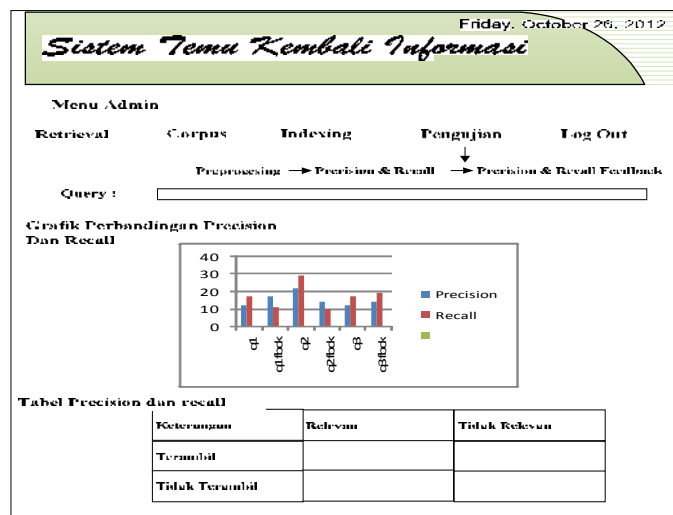
Gambar 4.15 Rancangan Menu Daftar *Corpus*

10. Tampilan Pengujian

Gambar 4.16 dan Gambar 4.17 adalah menu pengujian *precision* dan *recall* dari dokumen hasil pencarian, baik pencarian pertama atau pencarian kedua (hasil *feedback*). Menu ini juga memberikan tampilan tabel dan grafik perbandingan *precision* dan *recall* secara interpolasi seperti pada gambar 4.16 dan *non interpolasi* seperti Gambar 4.17 terhadap dokumen hasil pencarian yang telah dilakukan penilaian.



Gambar 4.16 Rancangan Menu Pengujian *Precision* dan *Recall* interpolasi



Gambar 4.17 Rancangan Menu Pengujian *Precision* dan *Recall* non-interpolasi

Pada Gambar 4.18 adalah menu pengujian simulasi *preprocessing*, untuk membuktikan proses pada *preprocessing* sistem temu kembali informasi telah berhasil dilakukan.

Tuesday, October 16, 2012

Sistem Temu Kembali Informasi

Menu Admin

Retrieval Corpus Indexing Pengujian Log Out

Preprocessing → Precision & Recall → Precision & Recall Feedback

Simulasi Preprocessing

Kata

Kata Inputan

Kategori Stopword

Kategori Tanda Baca

Kategori Stemming

Gambar 4.18 Rancangan Menu Pengujian *Preprocessing*

BAB V

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan menjelaskan tentang implementasi sistem temu kembali informasi sesuai dengan rancangan yang telah dilakukan pada bab sebelumnya. Bab ini juga akan membahas mengenai pengujian sistem, untuk mengetahui sejauh mana sistem temu kembali informasi yang telah dirancang berhasil dijalankan dan menguji *output* dari sistem temu kembali informasi.

5.1 Implementasi

Implementasi adalah tahapan dimana sistem telah selesai dianalisa dan dirancang sehingga sistem siap dioperasikan pada kondisi yang sebenarnya. Dari proses implementasi maka akan diketahui sejauh mana tingkat keberhasilan sistem terhadap analisa dan perancangan yang telah dilakukan. Kemudian dapat diketahui pula apakah sistem yang dibuat menghasilkan tujuan yang ingin dicapai.

5.1.1 Batasan Implementasi

Sistem temu kembali informasi yang dibangun pada tugas akhir memiliki batasan sebagai berikut :

1. Bahasa pemrograman yang digunakan dalam pengimplementasian sistem ini yaitu *Php* dengan DBMS *mySQL* pada sistem operasi *Microsoft Windows XP3*.
2. Tidak menggunakan kamus *stop word* dan *stemming* dalam bahasa asing, yang digunakan adalah bahasa indonesia sehingga tidak dapat melakukan pencarian yang mengandung kata bahasa asing.
3. Algoritma yang digunakan dalam pengembalian istilah ke kata dasar pada dokumen adalah Algoritma Nazief & Adriani.

4. Dalam penambahan koleksi dokumen, file yang dapat diinputkan adalah jurnal teknologi dan hanya file ekstensi *txt* dan dimasukan kedalam *DBMS mySQL*.

5.1.2 Lingkungan Operasional

Implementasi sistem temu kembali informasi ini dibagi kedalam dua komponen yaitu perangkat keras dan perangkat lunak, berikut ini adalah lingkungan operasional yang digunakan dalam pengimplementasian sistem:

a. Perangkat Keras

Processor : *Pentium® dual-core T2390, 1,86GHz*

Memori (RAM) : 2 GB

b. Perangkat Lunak

Sistem Operasi : *Microsoft Windows XP3.*

Bahasa Pemrograman : PHP

Web Browser : *Firefox*

DBMS : MySQL

Tools Perancangan : Notepad++

5.1.3 Hasil Implementasi

Pada subbab ini ditampilkan hasil implementasi sesuai dengan proses analisa dan perancangan pada bab sebelumnya yaitu berupa tampilan sistem yang dijalankan. Berikut adalah tampilan dari sistem temu kembali informasi:

1. Tampilan Menu Utama (Menu Pencarian)

Tampilan utama ini adalah sekaligus dengan tampilan menu pencarian. Ketika pengguna membuka sistem temu kembali informasi, maka tampilan utama yaitu menu pencarian akan muncul dan dapat di akses. Tampilan menu utam (menu pencarian) terlihat pada Gambar 5.1 berikut ini.

2. Tampilan Hasil Pencarian Menu Utama

Gambar 5.2 Hasil Pencarian Menu Utama

3. Tampilan Hasil Pencarian *feedback*

Tampilan pada Gambar 5.3 adalah hasil pencarian kedua, yaitu hasil pencarian kemiripan dokumen yang di *feedback* oleh pengguna pada hasil pencarian pertama.

The screenshot shows a web application titled "SISTEM TEMU KEMBALI INFORMASI" with a date "Thu, 30 10 12". Below the title is a "RETRIEVAL" section. It contains a search bar with the text "Silahkan inputkan keyword (sistem temu kembali)" and a "cari" button. Below the search bar, it says "Jumlah Dokumen Yang Di Temukan = 10". The first search result is for a document titled "NAMA DOKUMEN = CUSTOMER INFORMATION GATHERING MENGGUNAKAN METODE TEMU KEMBALI INFORMASI DENGAN MODEL RUANG VEKTOR" with a "NILAI FITNESS = 0.726". The second result is for "NAMA DOKUMEN = ANALISA DATA TRANSAKSIONAL PADA E-COMMERCE DENGAN" with a "NILAI FITNESS = 0.537". Each result includes a brief description of the document's content.

Gambar 5.3 Hasil Pencarian *Feedback*

4. Tampilan Menu *Login*

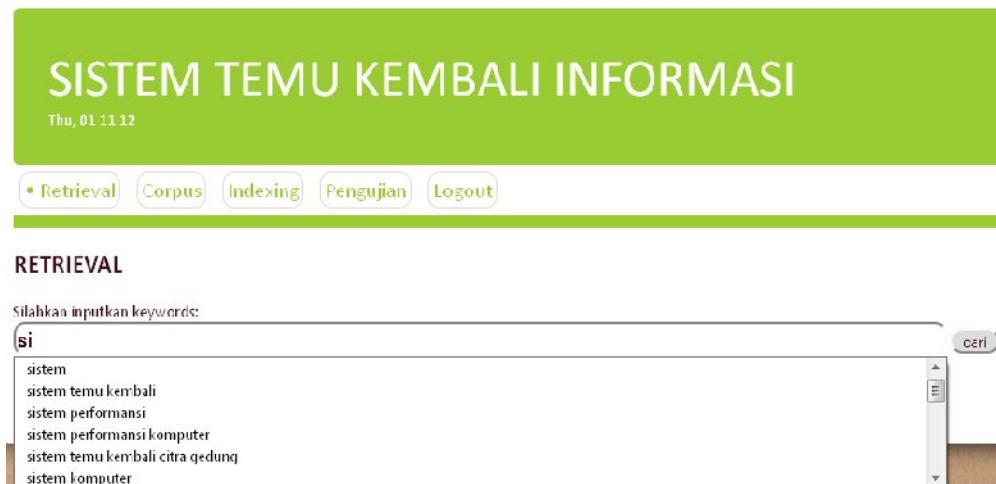
Tampilan Gambar 5.4 adalah menu *login* sebagai *admin*, yaitu dengan menginputkan *username* dan *password* kemudian *submit*. Maka setelah berhasil melakukan *login* maka hak akses pengguna terhadap sistem sebagai *admin*, yaitu mempunyai kebebasan dalam mengelola sistem.

The screenshot shows the login page of the "SISTEM TEMU KEMBALI INFORMASI" web application. It features a green header with the title and date "Thu, 01 11 12". Below the header is a login form with a "Login Admin" button. The form includes fields for "Username" and "Password", and a "Submit" button.

Gambar 5.4 Menu *Login* Sebagai *Admin*

5. Tampilan Menu Pencarian *Admin*

Tampilan pada Gambar 5.5 adalah menu pencarian *admin*. Menu ini sama dengan menu utama (pencarian pertama), yang membedakannya adalah menu ini hanya dapat diakses oleh *admin*.



Gambar 5.5 Menu Pencarian *Admin*

6. Tampilan Hasil Pencarian *Admin*

Tampilan pada Gambar 5.6 adalah hasil pencarian dari pencarian yang dilakukan oleh *admin*. Menu ini sama pada pencarian dokumen yang dilakukan oleh pengguna pada menu pencarian utama, hanya saja yang membedakan adalah pada menu ini hasil pencarian dokumen disertai info bobot perhitungan *similarity* dari proses *binary independence model*.

SISTEM TEMU KEMBALI INFORMASI

Thu, 03.12.22

[Retrieval](#)
[Corpus](#)
[Indexing](#)
[Pengujian](#)
[Logout](#)

RETRIEVAL

Silahkan inputkan keyword:

Binary

HASIL QUERY

1 Dokumentasi Hasil Pencarian
 Jumlah Dokumen yang ditemukan adalah 9 Dokumen

No Dokumen : 2
 Nama Dokumen : algoritma genetika untuk penyelesaian masalah vehicle routing
 Similarnya : 1.11613

materi ahli komputer agustus **genetika** penyelesaian masalah matematika universitas merupakan permasalahan pada sistem transportasi bertujuan meminimalkan total jarak biaya pengoperasian kendaraan minimal masalah ini masuk ke dalam pengetahuan mencari solusi ini **genetika** merupakan satu metode analog proses evolusi alam seleksi natural jika diartikan sebagai simulasi evolusi makhluk hidup dalam mencari solusi kasus di bandingkan menggunakan metode hasil berdasarkan solusi citra lain **genetika** mempunyai sifat jumlah individu pengoperasian perencanaan barang cukup kompleks variasi elemen jalur atau area biaya pengangkutan waktu pengangkutan permasalahan pendistribusian barang meminimalkan biaya pendistribusian mengambil rute kendaraan berangkat kembali pusat salah satu pengalaman riset operasi secara luas ahli raih tipe masalah digambarkan

Gambar 5.6 Hasil Pencarian *Binary*

7. Tampilan Hasil *Feedback Admin*

Tampilan pada Gambar 5.7 adalah hasil pencarian proses *feedback* yang dilakukan oleh *admin* untuk mendapatkan dokumen yang sama terhadap dokumen hasil pencarian pertama pada menu pencarian *admin*.

SISTEM TEMU KEMBALI INFORMASI

Thu, 03.12.22

[Retrieval](#)
[Corpus](#)
[Indexing](#)
[Pengujian](#)
[Logout](#)

RETRIEVAL

Silahkan inputkan keyword:

Feedback

Jumlah Dokumen yang ditemukan adalah 22 Dokumen

DOKUMEN 22
 NILAI FITNESS = 0.939
 NILAI CROSS OVER = 0.872
 NILAI MUTASI = 0.870
 Precision and Recall : /

materi ahli komputer agustus genetika penyelesaian masalah matematika universitas merupakan permasalahan pada sistem transportasi bertujuan meminimalkan total jarak biaya pengoperasian kendaraan minimal masalah ini masuk ke dalam pengetahuan mencari solusi ini genetika merupakan satu metode analog proses evolusi alam seleksi natural jika diartikan sebagai simulasi evolusi makhluk hidup dalam mencari solusi kasus di bandingkan menggunakan metode hasil berdasarkan solusi citra lain genetika mempunyai sifat jumlah individu pengoperasian perencanaan barang cukup kompleks variasi elemen jalur atau area biaya pengangkutan waktu pengangkutan permasalahan pendistribusian barang meminimalkan biaya pendistribusian mengambil rute kendaraan berangkat kembali pusat salah satu pengalaman riset operasi secara luas ahli raih tipe masalah digambarkan

Gambar 5.7 Hasil Pencarian *Feedback Admin*

8. Tampilan *Upload Corpus*

Tampilan pada Gambar 5.8 adalah implementasi menu *upload corpus*. Menu ini untuk mengelola koleksi dokumen jika ingin ditambah dengan dokumen yang baru kedalam *database*. *Admin* mengambil dimana lokasi penyimpanan dokumen yang ingin di *upload*, kemudian melakukan *upload*. Kemudian sistem akan melakukan proses *preprocessing* secara otomatis.



Gambar 5.8 *Upload Corpus*

9. Tampilan *Indexing*

Tampilan pada Gambar 5.9 adalah tampilan implementasi proses *indexing*. Semua dokumen yang di-*input* akan di-*index*. Tampilan menu *indexing* akan memberikan informasi mengenai keberhasilan *preprocessing*. Menu ini hanya dapat diakses oleh *admin*.

SISTEM TEMU KEMBALI INFORMASI

Thu, 01 11 12

[Retrieval](#)
[Corpus](#)
[• Indexing](#)
[Pengujian](#)
[Logout](#)

INDEXING

Total Dokumen = 20
 Total Kata = 30413
 Status Stem yang telah di Indexing = 6239

Dik: [Link berikut](#) ini untuk indexing Ulang

1st Dokumen | [View Detail](#)

No	Nama File	Jumlah Kata
1	algoritma elgama untuk enkripsi data menggunakan gnupg.txt	685
2	algoritma genetika untuk penyelesaian masalah vinidie routing.txt	786
3	algoritma umum pencarian informasi dalam sistem temu kembali berbasis metode vektorisasi kata dan dokumen.TXT	630
4	algoritman genetika untuk optimasi penjadwalan belajar mengajar.txt	1237
5	analisa algoritma dan waktu ekripsi versus deskripsi pada advanced encryption standard (aes).txt	1493
6	analisa cara menggunakan handphone sony ericson sebagai mouse melalui bluetooth.txt	1295
7	analisa data transaksi pada e-commerce dengan teknologi clap.txt	804
8	analisa teknik perbandingan router linux dengan router mikrotik pada jaringan wire ess.txt	1572
9	analisis dan perancangan sistem mobile xrs berbasis j2me menggunakan jaringan gprs.txt	804
10	analisis kelemahan keamanan pada jaringan wire ess.txt	1374

Gambar 5.9 Indexing

10. Tampilan Detail *Indexing*

Tampilan pada Gambar 5.10 adalah menu dari detail proses *indexing*, menu ini akan memberikan informasi daftar *corpus* yang tersimpan dalam *database* yang telah dilakukan *indexing* secara detail. Proses ini juga hanya dapat diakses oleh *admin*.

SISTEM TEMU KEMBALI INFORMASI

Sat, 27 10 12

[Retrieval](#)
[Corpus](#)
[• Indexing](#)
[Pengujian](#)
[Logout](#)

INDEXING

Total Dokumen = 20
 Total Kata = 17665
 Status Stem yang telah di Indexing = 5193

Dik: [Link berikut](#) ini untuk indexing Ulang

id	term	termddck	ic_dck	df	q+	pk	qlp+	logqlpk
361	etax	2	18	13	0.95	0.025	0.475	1.27875
362	etax	2	19	13	0.95	0.025	0.475	1.27875
363	etax	3	20	13	0.95	0.025	0.475	1.27875
364	etax	1	21	13	0.95	0.025	0.475	1.27875
365	etax	2	2	4	0.2	0.4	0.1	0.50206
367	etax	1	11	4	0.2	0.4	0.1	0.50206
367	etax	2	15	1	0.2	0.4	0.1	0.50206
368	etax	2	16	4	0.2	0.4	0.1	0.50206
369	etax	13	1	9	0.45	0.275	0.225	0.1871502
370	etax	1	2	9	0.45	0.275	0.225	0.1871502

Page: 1/1

Gambar 5.10 Hasil Detail *Indexing*

11. Tampilan Menu Pengujian

Pada menu pengujian ada tiga proses, proses yang pertama yaitu simulasi *preprocessing*, seperti pada tampilan Gambar 5.11. tampilan implementasi menu simulasi *preprocessing* ini untuk membuktikan bahwa proses dari *preprocessing* dalam sistem temu kembali ini telah berhasil digunakan.

SISTEM TEMU KEMBALI INFORMASI
Sat, 27 10 12

Retrieval Corpus Indexing **Pengujian** Logout

• Preprocessing Precision & Recall Precision & Recall Feedback

Simulasi Preprocessing

kata (

Data Inputan : dari mana saja kita berada ? dan dari situ kita ada !\n \n'; makan memakan termakan menyampaikan sampai tersampaikan

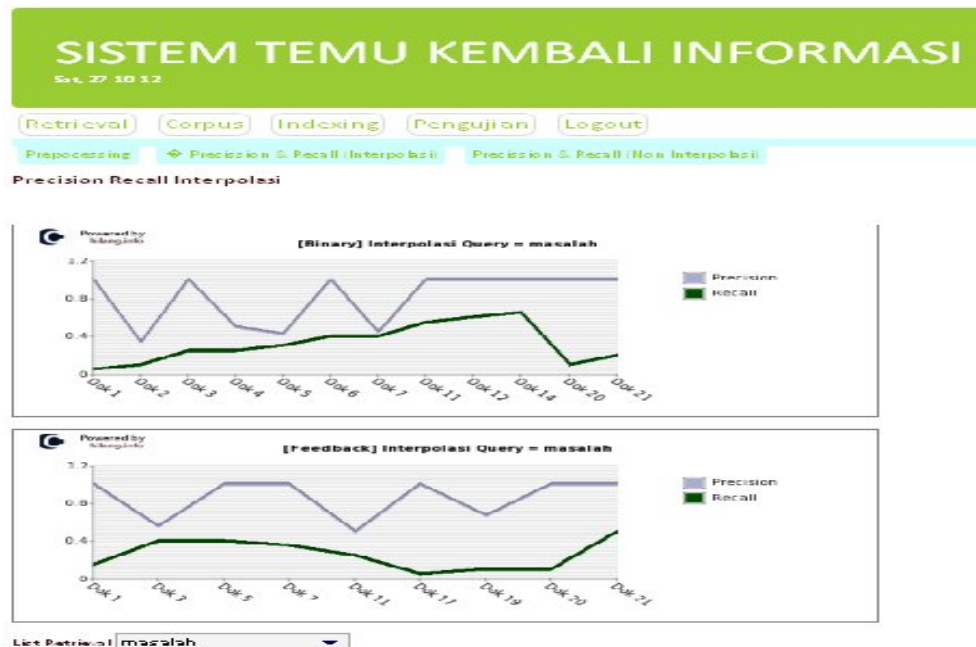
Stopword :
sampai ada kita dari dan kita saja mana dari

Tanda Baca :
?

stemming :
tersampaikan = sampai
menyampaikan = sampai
memakan = makan
memakan = makan
situ = situ
berada = ada

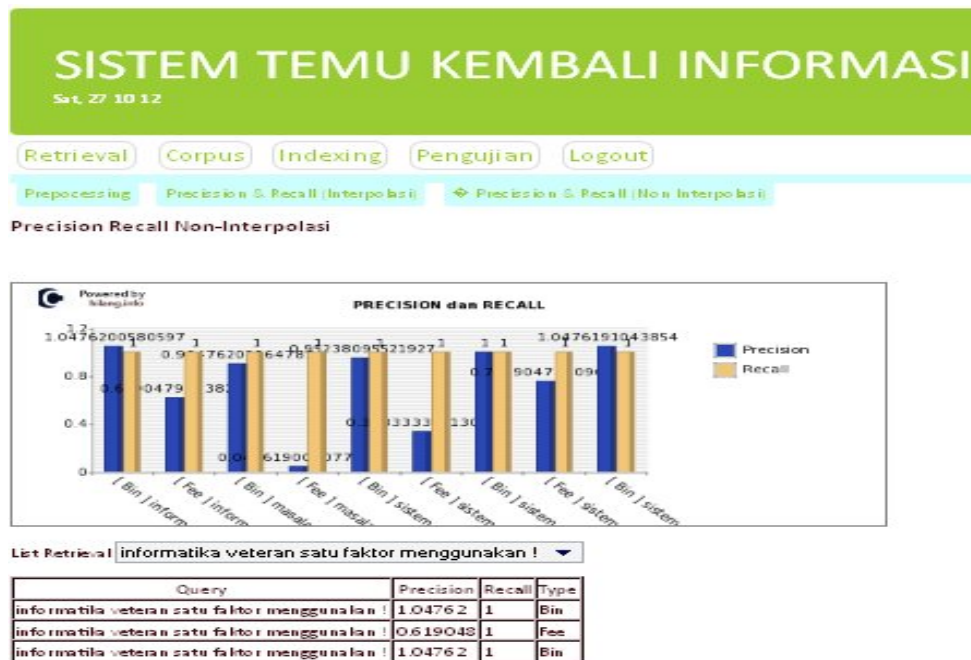
Gambar 5.11 Menu Pengujian Simulasi *Preprocessing*

Proses kedua pada menu pengujian ini yaitu pengujian nilai precision dan recall, seperti pada tampilan pada Gambar 5.12 adalah menu pengujian dari hasil pencarian sistem temu kembali informasi. Pada menu pengujian akan memberikan tampilan informasi berupa tabel pengujian *precision* dan *recall* secara interpolasi. Kemudian pengujian *precision* dan *recall* interpolasi tersebut akan konversi kedalam bentuk grafik.



Gambar 5.12 Menu Pengujian *Precision* dan *Recall Interpolasi*

Pada menu ini juga akan ditampilkan nilai *precision* dan *recall* non interpolasi hasil pencarian sistem temu kembali informasi. Kemudian juga menampilkan hasil konversinya berupa grafik. Hasil pengujian tersimpan sesuai dengan *query* yang dijadikan *input*-an seperti pada Gambar 5.13.



Gambar 5.13 Menu Pengujian *Precision* dan *Recall* Non-Interpolasi

5.2. Pengujian

Dalam tahapan ini, sistem akan dijalankan dan diuji cobakan untuk mengetahui apakah sistem berjalan sesuai dengan hasil analisa dan tujuan yang diharapkan. Untuk mengetahui kemampuan sistem yang telah dibangun, maka akan dilakukan pengujian dengan mengukur kualitas *retrieval*, yaitu dengan menghitung nilai *precision* dan *recall*.

5.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan adalah menguji apakah sistem yang telah dibangun dan diimplementasikan dapat memberikan hasil keluaran yang sesuai dengan yang diharapkan kemudian menghitung nilai *precision* dan *recall* (baik *interpolasi* atau *non-interpolasi*) dari hasil pencarian sistem temu kembali informasi. Beberapa langkah yang akan dilakukan dalam pengujian, yaitu sebagai berikut:

1. Menilai kemampuan sistem dalam me-*retrieve* dokumen yang relevan dengan *query* pengguna dengan cara menghitung nilai *precision* dan *recall* (baik *interpolasi* atau *non-interpolasi*) sebagai parameter kualitas *retrieval* sistem temu kembali informasi dengan *binary independence model* terhadap sejumlah *query* yang berbeda panjang pendeknya kata pada *query* ke dalam sistem temu kembali informasi.
2. Menilai kemampuan sistem dalam me-*retrieve* dokumen yang relevan dengan dokumen yang di jadikan *feedback* oleh pengguna dengan cara menghitung nilai *precision* dan *recall* (baik *interpolasi* atau *non-interpolasi*) sebagai parameter kualitas metode *relevance feedback* dengan algoritma genetika terhadap dokumen yang di-*feedback* oleh pengguna berupa dokumen yang relevan dan dokumen yang tidak relevan.
3. Membandingkan nilai *precision* dan *recall* (baik *interpolasi* atau *non-interpolasi*) hasil pencarian dari *binary independence model* dan *relevance feedback*.

5.2.2 Lingkungan Pengujian

Pengujian sistem temu kembali informasi ini dibagi kedalam dua komponen yaitu perangkat keras dan perangkat lunak, berikut ini adalah lingkungan operasional yang digunakan dalam pengujian sistem:

a. Perangkat Keras

Processor : *Pentium® dual-core T2390, 1,86GHz*

Memori (RAM) : 2 GB

b. Perangkat Lunak

Sistem Operasi : *Microsoft Windows XP3.*

Bahasa Pemrograman : PHP

Web Browser : *Firefox*

DBMS : MySQL

5.2.3 Hasil Pengujian Unjuk Kerja Sistem

Berikut adalah tabel *query* yang akan dijadikan pengujian:

Tabel 5.1: Daftar *query* Pengujian

No	<i>Query</i> Pengujian	Jenis Pengujian		Jumlah Dokumen Yang Ditemukan Sistem	
		<i>Model Binary Independence</i>	<i>Relevance Feedback</i>	<i>Model Binary Independence</i>	<i>Relevance Feedback</i>
1	Genetika	<i>Query</i> Pendek	Dokumen Relevan	7 Dokumen	4 Dokumen
2	Enkripsi	<i>Query</i> Pendek	Dokumen Tidak Relevan	24 Dokumen	17 Dokumen
3	Kecerdasan Buatan	<i>Query</i> Panjang	Dokumen Relevan	11 Dokumen	20 Dokumen
4	Klasifikasi Keberadaan Kalimat	<i>Query</i> Panjang	Dokumen Tidak Relevan	21 Dokumen	17 Dokumen
5	Pendeteksi Penyakit	<i>Query</i> Panjang	Dokumen Tidak Relevan	25 Dokumen	19 Dokumen
6	Jaringan Saraf Tiruan	<i>Query</i> Panjang	Dokumen Tidak Relevan	25 Dokumen	19 Dokumen

Setelah *query* diinputkan, sistem akan menghitung relevansi dengan koleksi dokumen (corpus) berdasarkan perhitungan rumus *recall* dan *precision interpolasi* pada rumus 2.1 dan rumus 2.2. Berikut adalah hasil pengujian dari *query* yang ada pada Tabel 5.1 di atas:

1. Hasil Pengujian *Precision* dan *Recall Interpolasi* Terhadap Kata Yang Pendek Untuk *Query*1

Pengujian dengan menginputkan kata yang pendek sebagai *query* dilakukan dengan menginputkan *query* yang satu suku kata. Tujuan dari pengujian

ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall interpolasi* dari hasil pencarian yang menginputkan *query* yang pendek (satu suku kata). Tabel 5.2 berikut adalah hasil pengujian untuk *query* “Genetika”:

Tabel 5.2: Hasil Pengujian Interpolasi untuk Query 1: Genetika

No	No Dokumen	Relevan?	Precision (P)	Recall (R).
1	2	Ya	$P = 1/1 = 1$	$R = 1/7 = 0.142$
2	22	Ya	$P = 2/2 = 1$	$R = 2/7 = 0.285$
3	57	Ya	$P = 3/3 = 1$	$R = 3/7 = 0.428$
4	59	Ya	$P = 4/4 = 1$	$R = 4/7 = 0.571$
5	68	Ya	$P = 5/5 = 1$	$R = 5/7 = 0.714$
6	72	Ya	$P = 6/6 = 1$	$R = 6/7 = 0.857$
7	88	Ya	$P = 7/7 = 1$	$R = 7/7 = 1$

2. Hasil Pengujian Precision dan Recall Interpolasi Dari Relevance Feedback Terhadap Query Pendek dan Dokumen Yang Dijadikan Feedback Adalah Dokumen Yang Relevan

Setelah hasil pencarian telah berhasil ditemukan, pengguna akan melakukan *feedback* kedalam sistem dengan memilih salah satu dokumen yang akan dijadikan *feedback*. Dokumen yang dipilih untuk dilakukan *feedback* adalah dokumen yang relevan yaitu nomor dokumen 22 dengan judul dokumen: “Clustering Untuk Peningkatan Efektifitas Penyajian Informasi Dari Mesin Pencari Teks”.

Tujuan dari memberikan dokumen yang relevan sebagai *feedback* adalah menilai kemampuan metode *relevance feedback* dengan algoritma genetika dalam menemukan hasil pencarian terhadap dokumen yang relevan sebagai *feedback*. Penilaian dilakukan dengan menilai *precision* dan *recall interpolasi* dari hasil

pencarian proses *relevance feedback*. Maka hasil pengujian *relevance feedback* dengan menggunakan algoritma genetika terlihat pada tabel 5.3 berikut:

Tabel 5.3: Hasil Pengujian Interpolasi *Feedback* untuk No. Dokumen 2

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	25	Ya	$P = 1/1 = 1$	$R = 1/4 = 0.25$
2	27	Ya	$P = 2/2 = 1$	$R = 2/4 = 0.5$
3	60	Ya	$P = 3/3 = 1$	$R = 3/4 = 0.75$
4	78	Ya	$P = 4/4 = 1$	$R = 4/4 = 1$

3. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Terhadap Kata Yang Pendek Untuk *Query* 1

Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall non-interpolasi* dari hasil pencarian yang menginputkan *query* yang pendek (satu suku kata).

Berdasarkan Tabel 5.2, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 7 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 0 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 93 dokumen.

Tabel 5.4. Hasil pengujian *Precision* dan *Recall Non-interpolasi* pada Q_1

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	7 (<i>tp</i>)	0 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	93 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *query* 1 adalah:

$$Precision \rightarrow P = tp / (tp + fp) = 7 / (7+0) = 7/7 = 1$$

$$Recall \rightarrow R = tp / (tp + fn) = 7 / (7+0) = 7/7 = 1$$

4. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Dari *Relevance Feedback* Terhadap *Query* Pendek dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Relevan

Dari proses *feedback* yang dilakukan maka, berdasarkan Tabel 5.3, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 4 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 0 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 96 dokumen.

Tujuan pengujian ini untuk menilai *precision* dan *recall non-interpolasi* dari hasil pencarian proses *relevance feedback* dengan algoritma genetika terhadap *query* pendek dan dokumen yang dijadikan *feedback* adalah dokumen yang relevan. Tabel 5.5 berikut adalah hasil pengujiannya:

Tabel 5.5. Hasil pengujian *Feedback Precision* dan *Recall Non-interpolasi* pada Q_1

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	4 (<i>tp</i>)	0 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	96 (<i>tn</i>)

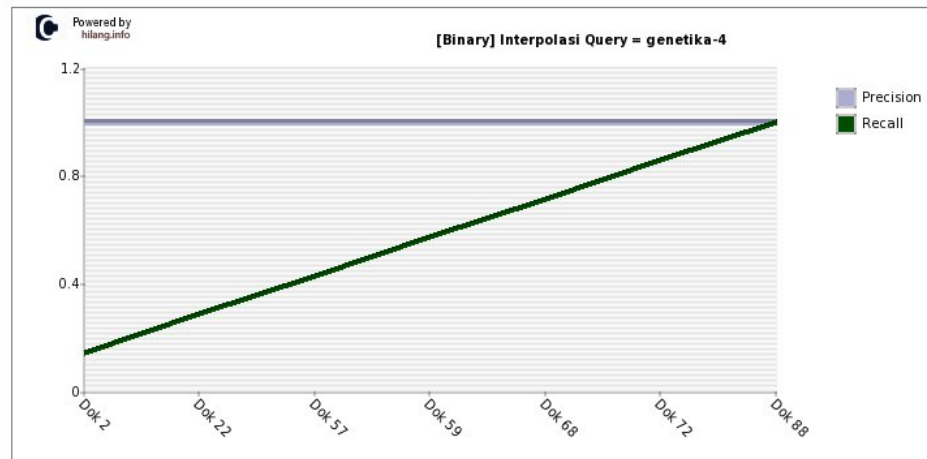
Maka, nilai *precision* dan *recall* untuk *feedback* adalah:

$$Precision \rightarrow P = tp / (tp + fp) = 4 / (4+0) = 4/4 = 1$$

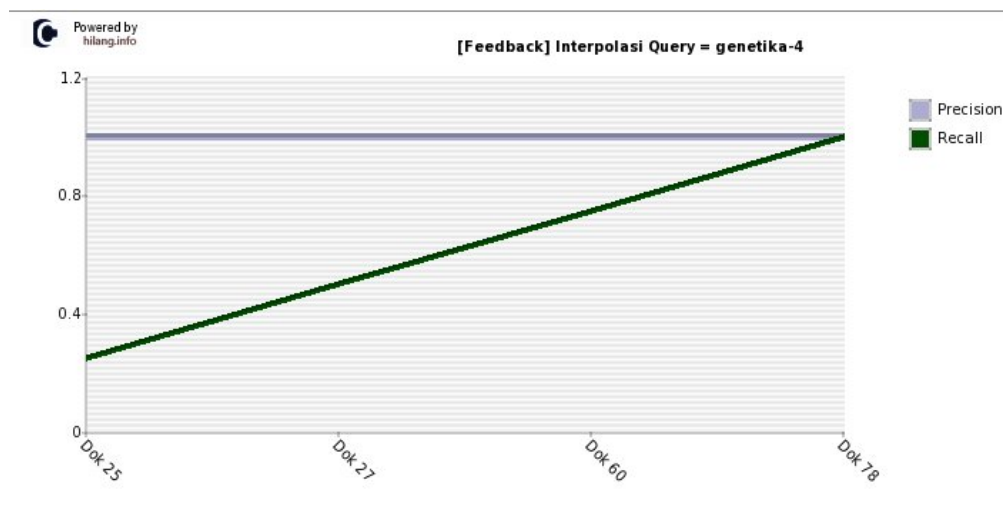
$$Recall \rightarrow R = tp / (tp + fn) = 4 / (4+0) = 4/4 = 1$$

Berikut adalah gambar grafik perbandingan *precision* dan *recall* secara interpolasi, *metode binary independence* dan *relevance feedback*:

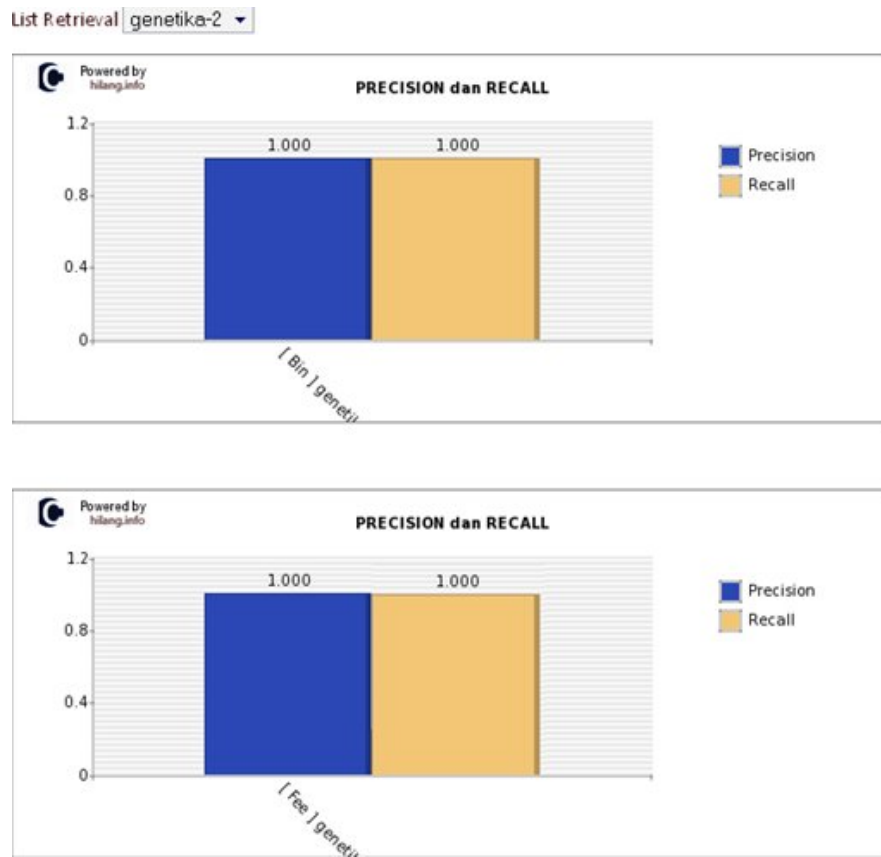
List Retrieval genetika-4



Gambar 5.14 Hasil Pengujian *precision* dan *Recall* interpolasi untuk Query "Genetika"



Gambar 5.15 Hasil Pengujian *Precision* dan *Recall* interpolasi untuk *Feedback* Dokumen 22



Gambar 5.16 Grafik Pengujian *Precision* dan *Recall* Non-interpolasi untuk Query "Genetika"

1. Hasil Pengujian *Precision* dan *Recall* Interpolasi Terhadap Kata Yang Pendek Untuk Query 2

Pengujian dengan menginputkan kata yang pendek sebagai *query* dilakukan dengan menginputkan *query* yang satu suku kata. Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall* interpolasi dari hasil pencarian yang menginputkan *query* yang pendek (satu suku kata). Hasil perhitungan *recall* dan *precision* interpolasi untuk pengujian kedua dapat dilihat pada Tabel 5.6:

Tabel 5.6: Hasil Pengujian Interpolasi untuk *Query 2*: Enkripsi

No	No Dokumen	Relevan?	<i>Precision (P)</i>	<i>Recall (R).</i>
1	1	Ya	$P = 1/1 = 1$	$R = 1/20 = 0.05$
2	73	Ya	$P = 2/2 = 1$	$R = 2/20 = 0.1$
3	5	Ya	$P = 3/3 = 1$	$R = 3/20 = 0.15$
4	80	Ya	$P = 4/4 = 1$	$R = 4/20 = 0.2$
5	6	Tidak	$P = 4/5 = 0.8$	$R = 4/20 = 0.2$
6	81	Ya	$P = 5/6 = 0.833$	$R = 5/20 = 0.25$
7	10	Ya	$P = 6/7 = 0.857$	$R = 6/20 = 0.3$
8	82	Ya	$P = 7/8 = 0.875$	$R = 7/20 = 0.35$
9	14	Ya	$P = 8/9 = 0.888$	$R = 8/20 = 0.4$
10	91	Ya	$P = 9/10 = 0.9$	$R = 9/20 = 0.45$
11	15	Ya	$P = 10/11 = 0.909$	$R = 10/20 = 0.5$
12	92	Ya	$P = 11/12 = 0.916$	$R = 11/20 = 0.55$
13	21	Ya	$P = 12/13 = 0.923$	$R = 12/20 = 0.6$
14	93	Ya	$P = 13/14 = 0.928$	$R = 13/20 = 0.65$
15	24	Tidak	$P = 13/15 = 0.866$	$R = 13/20 = 0.65$
16	94	Ya	$P = 14/16 = 0.875$	$R = 14/20 = 0.7$
17	26	Ya	$P = 15/17 = 0.882$	$R = 15/20 = 0.75$
18	29	Ya	$P = 16/18 = 0.888$	$R = 16/20 = 0.8$
19	30	Ya	$P = 17/19 = 0.894$	$R = 17/20 = 0.85$
20	31	Ya	$P = 18/20 = 0.9$	$R = 18/20 = 0.9$
21	40	Ya	$P = 19/21 = 0.904$	$R = 19/20 = 0.95$
22	50	Tidak	$P = 19/22 = 0.863$	$R = 19/20 = 0.95$
23	64	Tidak	$P = 19/23 = 0.826$	$R = 19/20 = 0.95$
24	70	Ya	$P = 20/24 = 0.83$	$R = 20/20 = 1$

2. Hasil Pengujian *Precision* dan *Recall* Interpolasi Dari *Relevance Feedback* Terhadap *Query* Pendek dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Tidak Relevan

Dokumen yang dipilih untuk dilakukan *feedback* adalah dokumen yang tidak relevan yaitu nomor dokumen 24 dengan judul dokumen: “**Desain Implementasi Sistem Komunikasi Wireless Pada Sbc Alix**”. Tujuannya adalah menilai *precision* dan *recall* interpolasi dari *relevance feedback* dengan algoritma genetika jika dokumen yang dijadikan *feedback* adalah dokumen yang tidak relevan. Hasil pengujian *relevance feedback* dari dokumen yang tidak relevan yang dijadikan *feedback* dapat dilihat pada Tabel 5.7 sebagai berikut:

Tabel 5.7: Hasil Pengujian Interpolasi *Feedback* untuk No. Dokumen 24

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	86	Tidak	$P = 0/1 = 0$	$R = 0/13 = 0$
2	18	Tidak	$P = 0/2 = 0$	$R = 0/13 = 0$
3	1	Ya	$P = 1/3 = 0.333$	$R = 1/13 = 0.076$
4	53	Tidak	$P = 1/4 = 0.25$	$R = 1/13 = 0.076$
5	49	Ya	$P = 2/5 = 0.4$	$R = 2/13 = 0.153$
6	39	Ya	$P = 3/6 = 0.5$	$R = 3/13 = 0.23$
7	98	Ya	$P = 4/7 = 0.571$	$R = 4/13 = 0.307$
8	29	Ya	$P = 5/8 = 0.625$	$R = 5/13 = 0.384$
9	92	Ya	$P = 6/9 = 0.666$	$R = 6/13 = 0.461$
10	63	Ya	$P = 7/10 = 0.7$	$R = 7/13 = 0.538$
11	69	Ya	$P = 8/11 = 0.727$	$R = 8/13 = 0.615$
12	90	Ya	$P = 9/12 = 0.75$	$R = 9/13 = 0.692$
13	79	Ya	$P = 10/13 = 0.769$	$R = 10/13 = 0.769$
14	8	Ya	$P = 11/14 = 0.785$	$R = 11/13 = 0.846$
15	73	Ya	$P = 12/15 = 0.8$	$R = 12/13 = 0.923$
16	13	Ya	$P = 13/16 = 0.812$	$R = 13/13 = 1$
17	45	Tidak	$P = 13/17 = 0.764$	$R = 13/13 = 1$

3. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Terhadap Kata Yang Pendek Untuk *Query 2*

Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall non-interpolasi* dari hasil pencarian yang menginputkan *query* yang pendek (satu suku kata).

Berdasarkan Tabel 5.6, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query (tp)* sebanyak 20 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 4 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query (fn)* sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 76 dokumen. Berikut pengujian menggunakan *presicion* dan *recall* yang tidak terinterpolasi.

Tabel 5.8. Hasil Pengujian *Presicion* dan *Recall Non-interpolasi* pada *Q₂*

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	20 (<i>tp</i>)	4 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	76 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *query 1* adalah:

$$Precision \rightarrow P = tp / (tp + fp) = 20 / (20+4) = 20/24 = 0.833$$

$$Recall \rightarrow R = tp / (tp + fn) = 20 / (20+0) = 20/20 = 1$$

4. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Dari *Relevance Feedback* Terhadap *Query* Pendek dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Tidak Relevan.

Dari proses *feedback* yang dilakukan maka, berdasarkan Tabel 5.7, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query (tp)* sebanyak 13 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 4 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang

relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 83 dokumen.

Tujuan pengujian ini untuk menilai *precision* dan *recall non-interpolasi* dari hasil pencarian proses *relevance feedback* dengan algoritma genetika terhadap *query* pendek dan dokumen yang dijadikan *feedback* adalah dokumen yang tidak relevan. Berikut adalah hasil pengujiannya:

Tabel 5.9. Hasil Pengujian *Feedback Presicion* dan *Recall Non-inperpolasi* pada Q₂

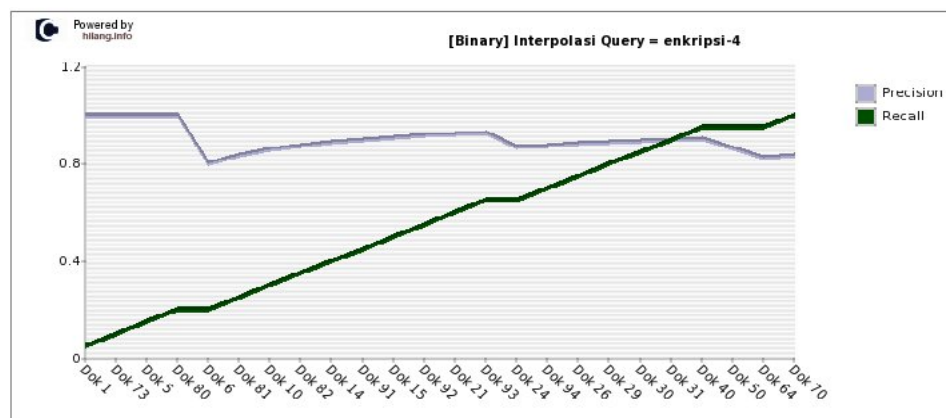
	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	13 (<i>tp</i>)	4 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	83 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *feedback* adalah:

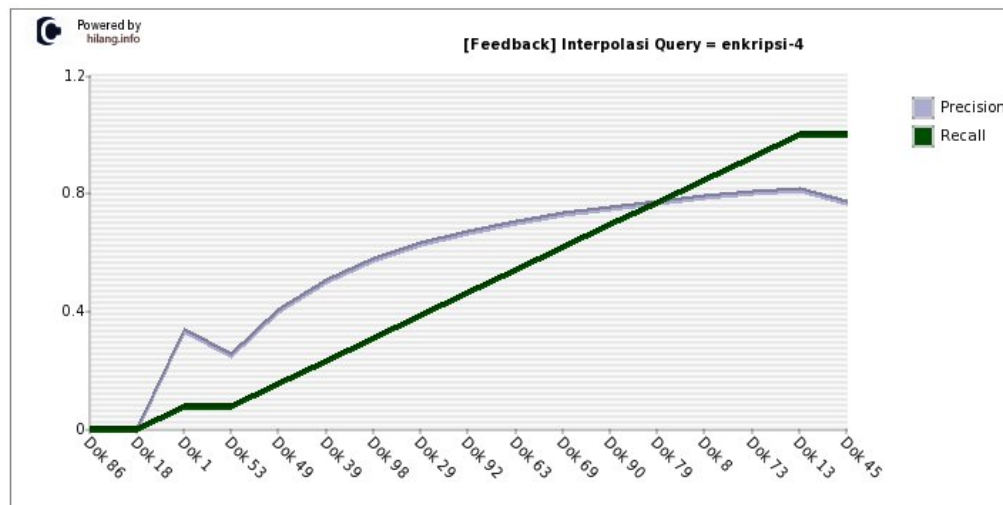
$$\text{Precision} \rightarrow P = tp / (tp + fp) = 13 / (13+4) = 13/17 = 0.764$$

$$\text{Recall} \rightarrow R = tp / (tp + fn) = 13 / (13+0) = 13/13 = 1$$

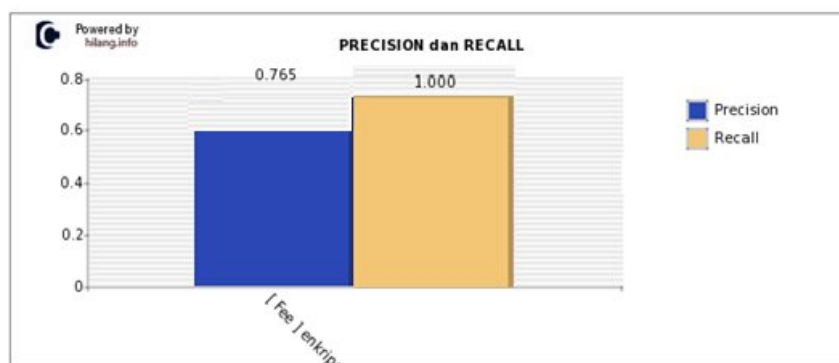
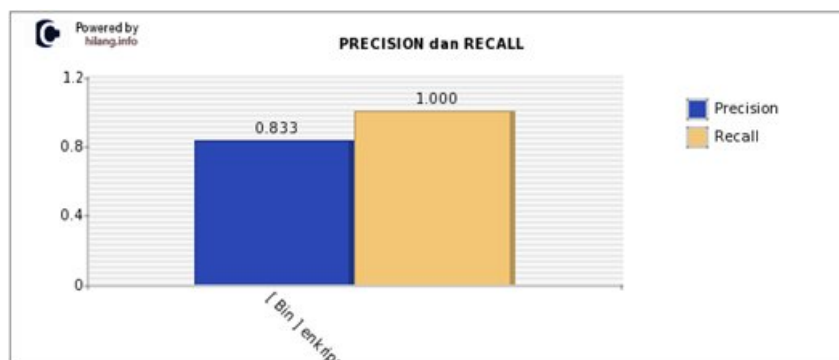
Berikut adalah grafik *precision* dan *recall interpolasi query 2*:



Gambar 5.17 Hasil Pengujian *precision* dan *Recall interpolasi* untuk *Query* "Enkripsi"



Gambar 5.18 Hasil Pengujian *precision* dan *Recall* interpolasi untuk *Feedback* Dokumen 24



Gambar 5.19 Hasil Pengujian *Precision* dan *Recall* non-interpolasi untuk *Query* "Enkripsi"

1. Hasil Pengujian *Precision* dan *Recall Interpolasi* Terhadap Kata Yang Panjang Untuk *Query 3*

Pengujian dengan menginputkan kata yang panjang sebagai *query* dilakukan dengan menginputkan *query* yang dua suku kata. Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (dua suku kata). Hasil perhitungan *recall* dan *presicion interpolasi* untuk pengujian ke ketiga seperti pada Tabel 5.10 berikut:

Tabel 5.10: Hasil Pengujian Interpolasi untuk *Query 3*: Kecerdasan Buatan

No	No Dokumen	Relevan?	<i>Precision (P)</i>	<i>Recall (R).</i>
1	12	Ya	$P = 1/1 = 1$	$R = 1/8 = 0.125$
2	13	Ya	$P = 2/2 = 1$	$R = 2/8 = 0.25$
3	18	Ya	$P = 3/3 = 1$	$R = 3/8 = 0.375$
4	28	Ya	$P = 4/4 = 1$	$R = 4/8 = 0.5$
5	60	Tidak	$P = 4/5 = 0.8$	$R = 4/8 = 0.5$
6	65	Tidak	$P = 4/6 = 0.666$	$R = 4/8 = 0.5$
7	75	Tidak	$P = 4/7 = 0.571$	$R = 4/8 = 0.5$
8	77	Ya	$P = 5/8 = 0.625$	$R = 5/8 = 0.625$
9	87	Ya	$P = 6/9 = 0.666$	$R = 6/8 = 0.75$
10	88	Ya	$P = 7/10 = 0.7$	$R = 7/8 = 0.875$
11	89	Ya	$P = 8/11 = 0.727$	$R = 8/8 = 1$

2. Hasil Pengujian *Precision* dan *Recall Interpolasi* Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Relevan.

Dokumen yang dipilih untuk dilakukan *feedback* adalah dokumen yang relevan yaitu nomor dokumen 12 dengan judul dokumen: “**Aplikasi Diagnosis Penyakit Menggunakan Perangkat Bergerak Dengan Sistem Aplikasi Android**”. Tujuannya adalah menilai *precision* dan *recall intepolasi* dari *relevance*

feedback dengan algoritma genetika jika dokumen yang dijadikan *feedback* adalah dokumen yang relevan. Hasil pengujian *relevance feedback* dari dokumen yang relevan yang dijadikan *feedback* seperti pada Tabel 5.11 berikut:

Tabel 5.11: Hasil Pengujian Interpolasi *Feedback* untuk No. Dokumen 12

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	87	Ya	$P = 1/1 = 1$	$R = 1/7 = 0.142$
2	28	Ya	$P = 2/2 = 1$	$R = 2/7 = 0.285$
3	1	Tidak	$P = 2/3 = 0.666$	$R = 2/7 = 0.285$
4	81	Tidak	$P = 2/4 = 0.5$	$R = 2/7 = 0.285$
5	62	Tidak	$P = 2/5 = 0.4$	$R = 2/7 = 0.285$
6	61	Tidak	$P = 2/6 = 0.333$	$R = 2/7 = 0.285$
7	101	Tidak	$P = 2/7 = 0.285$	$R = 2/7 = 0.285$
8	8	Tidak	$P = 2/8 = 0.25$	$R = 2/7 = 0.285$
9	25	Tidak	$P = 2/9 = 0.222$	$R = 2/7 = 0.285$
10	16	Ya	$P = 3/10 = 0.3$	$R = 3/7 = 0.428$
11	13	Ya	$P = 4/11 = 0.363$	$R = 4/7 = 0.571$
12	36	Tidak	$P = 4/12 = 0.333$	$R = 4/7 = 0.571$
13	95	Ya	$P = 5/13 = 0.384$	$R = 5/7 = 0.714$
14	85	Tidak	$P = 5/14 = 0.357$	$R = 5/7 = 0.714$
15	72	Tidak	$P = 5/15 = 0.333$	$R = 5/7 = 0.714$
16	67	Ya	$P = 6/16 = 0.375$	$R = 6/7 = 0.857$
17	31	Tidak	$P = 6/17 = 0.352$	$R = 6/7 = 0.857$
18	49	Tidak	$P = 6/18 = 0.333$	$R = 6/7 = 0.857$
19	42	Ya	$P = 7/19 = 0.368$	$R = 7/7 = 1$
20	53	Tidak	$P = 7/20 = 0.35$	$R = 7/7 = 1$

3. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Terhadap Kata Yang Panjang Untuk *Query 3*

Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall non-interpolasi* dari hasil

pencarian yang menginputkan *query* yang panjang (dua suku kata). Maka berdasarkan Tabel 5.10, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 8 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 3 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 89 dokumen. Tabel 5.12 berikut adalah hasil pengujian menggunakan *presicion* dan *recal non-terinterpolasi*.

Tabel 5.12. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* pada Q3

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	7 (<i>tp</i>)	4 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	89 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *query* 1 adalah:

$$\text{Precision} \rightarrow P = tp / (tp + fp) = 7 / (7+3) = 7/11 = 0.636$$

$$\text{Recall} \rightarrow R = tp / (tp + fn) = 8 / (8+0) = 8/8 = 1$$

4. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Relevan.

Dari proses *feedback* yang dilakukan maka, berdasarkan Tabel 5.11, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 7 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 13 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 80 dokumen. Tujuan pengujian ini untuk menilai *precision* dan *recall non-interpolasi* dari hasil pencarian proses *relevance feedback* dengan algoritma genetika terhadap *query* panjang dan dokumen yang dijadikan *feedback* adalah dokumen yang relevan. Tabel 5.13 berikut adalah hasil pengujiannya:

Tabel 5.13. Hasil pengujian *Feedback Presicion* dan *Recall Non-interpolasi* pada Q_3

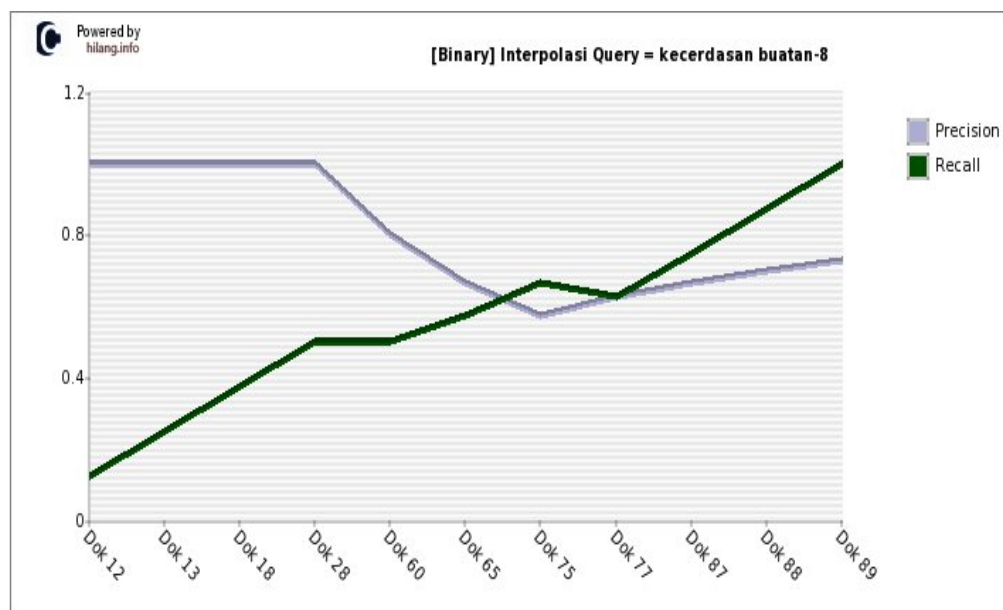
	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	7 (<i>tp</i>)	13 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	80 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *feedback* adalah:

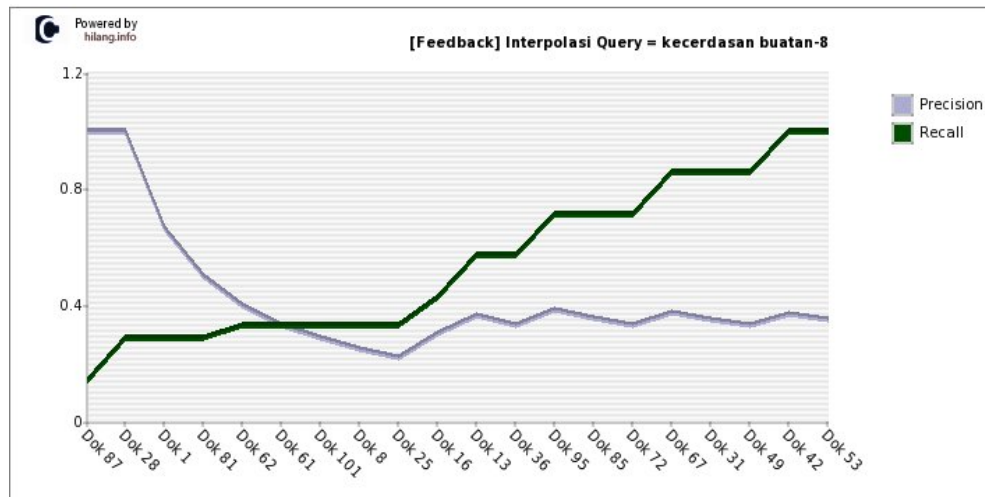
$$\text{Precision} \rightarrow P = tp / (tp + fp) = 7 / (7+13) = 7/20 = 0.35$$

$$\text{Recall} \rightarrow R = tp / (tp + fn) = 7 / (7+0) = 7/7 = 1$$

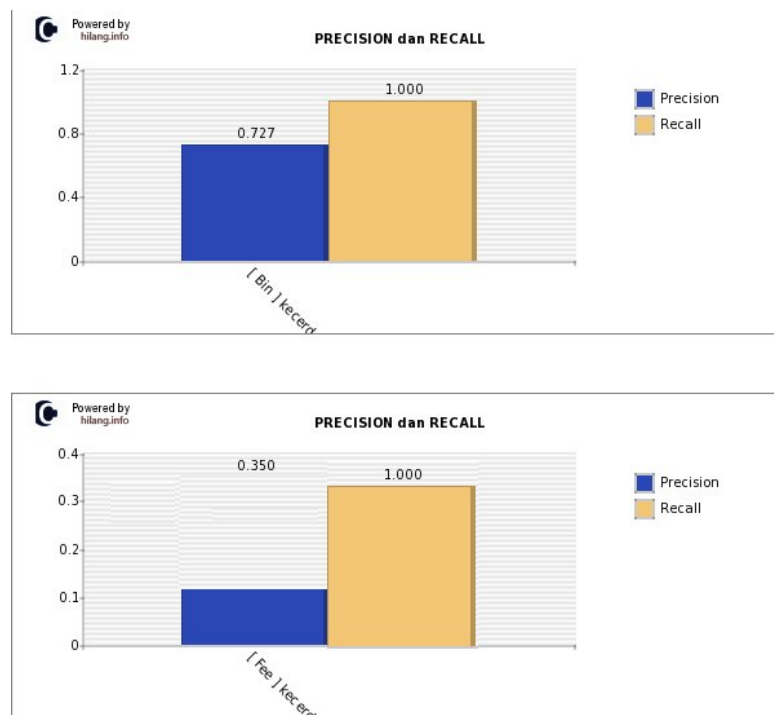
Berikut adalah grafik *precision* dan *recall* interpolasi *query* 3:



Gambar 5.20 Hasil Pengujian *precision* dan *Recall interpolasi* untuk Query "Kecerdasan Buatan"



Gambar 5.21 Hasil Pengujian *precision* dan *Recall* interpolasi untuk *Feedback* Dokumen 12



Gambar 5.22 Hasil Pengujian *precision* dan *Recall* non-interpolasi untuk Query "Kecerdasan Buatan"

1. Hasil Pengujian *Precision* dan *Recall Interpolasi* Terhadap Kata Yang Panjang Untuk *Query 4*: Klasifikasi Keberadaan Kalimat

Pengujian dengan menginputkan kata yang panjang sebagai *query*. Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (tiga suku kata). Tabel 5.14 adalah hasilnya:

Tabel 5.14: Hasil Pengujian *Precision* dan *Recall Interpolasi query 4*

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	25	Ya	$P = 1/1 = 1$	$R = 1/9 = 0.111$
2	27	Ya	$P = 2/2 = 1$	$R = 2/9 = 0.222$
3	28	Tidak	$P = 2/3 = 0.666$	$R = 2/9 = 0.222$
4	47	Ya	$P = 3/4 = 0.75$	$R = 3/9 = 0.333$
5	55	Ya	$P = 4/5 = 0.8$	$R = 4/9 = 0.444$
6	60	Ya	$P = 5/6 = 0.833$	$R = 5/9 = 0.555$
7	80	Tidak	$P = 5/7 = 0.714$	$R = 5/9 = 0.555$
8	90	Ya	$P = 6/8 = 0.75$	$R = 6/9 = 0.666$
9	91	Tidak	$P = 6/9 = 0.666$	$R = 6/9 = 0.666$
10	68	Tidak	$P = 6/10 = 0.6$	$R = 6/9 = 0.666$
11	69	Tidak	$P = 6/11 = 0.545$	$R = 6/9 = 0.666$
12	86	Tidak	$P = 6/12 = 0.5$	$R = 6/9 = 0.666$
13	87	Tidak	$P = 6/13 = 0.461$	$R = 6/9 = 0.666$
14	88	Tidak	$P = 6/14 = 0.428$	$R = 6/9 = 0.666$
15	20	Tidak	$P = 6/15 = 0.4$	$R = 6/9 = 0.666$
16	32	Ya	$P = 7/16 = 0.437$	$R = 7/9 = 0.777$
17	44	Ya	$P = 8/17 = 0.47$	$R = 8/9 = 0.888$
18	48	Ya	$P = 9/18 = 0.5$	$R = 9/9 = 1$
19	53	Tidak	$P = 9/19 = 0.473$	$R = 9/9 = 1$
20	56	Tidak	$P = 9/20 = 0.45$	$R = 9/9 = 1$
21	67	Tidak	$P = 9/21 = 0.428$	$R = 9/9 = 1$

2. Hasil Pengujian *Precision* dan *Recall* Interpolasi Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Tidak Relevan

Dokumen yang dipilih untuk dilakukan *feedback* adalah dokumen yang tidak relevan yaitu nomor dokumen 28 dengan judul dokumen: “**Deteksi Kerusakan Notebook Dengan Menggunakan Sistem Pakar**”. Tujuannya adalah menilai *precision* dan *recall* interpolasi dari *relevance feedback* dengan algoritma genetika jika dokumen yang dijadikan *feedback* adalah dokumen yang tidak relevan. Hasil pengujian *relevance feedback* dari dokumen yang tidak relevan yang dijadikan *feedback* seperti pada Tabel 5.15 berikut:

Tabel 5.15: Hasil Pengujian *Interpolasi Feedback* untuk No. Dokumen 28

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	87	Ya	$P = 1/1 = 1$	$R = 1/4 = 0.25$
2	11	Ya	$P = 2/2 = 1$	$R = 2/4 = 0.5$
3	85	Tidak	$P = 2/3 = 0.666$	$R = 2/4 = 0.5$
4	67	Tidak	$P = 2/4 = 0.5$	$R = 2/4 = 0.5$
5	49	Tidak	$P = 2/5 = 0.4$	$R = 2/4 = 0.5$
6	6	Tidak	$P = 2/6 = 0.333$	$R = 2/4 = 0.5$
7	93	Tidak	$P = 2/7 = 0.285$	$R = 2/4 = 0.5$
8	80	Tidak	$P = 2/8 = 0.25$	$R = 2/4 = 0.5$
9	65	Tidak	$P = 2/9 = 0.222$	$R = 2/4 = 0.5$
10	58	Tidak	$P = 2/10 = 0.2$	$R = 2/4 = 0.5$
11	54	Tidak	$P = 2/11 = 0.181$	$R = 2/4 = 0.5$
12	41	Tidak	$P = 2/12 = 0.166$	$R = 2/4 = 0.5$
13	36	Tidak	$P = 2/13 = 0.153$	$R = 2/4 = 0.5$
14	31	Tidak	$P = 2/14 = 0.142$	$R = 2/4 = 0.5$
15	22	Ya	$P = 3/15 = 0.2$	$R = 3/4 = 0.75$
16	19	Tidak	$P = 3/16 = 0.187$	$R = 3/4 = 0.75$
17	97	Ya	$P = 4/17 = 0.235$	$R = 4/4 = 1$

3. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Terhadap Kata Yang Panjang Untuk *Query 4*

Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall non-interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (tiga suku kata). Maka berdasarkan Tabel 5.14, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 9 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 12 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 79 dokumen. Tabel 5.16 berikut adalah hasil pengujiannya:

Tabel 5.16. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* pada Q4

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	9 (<i>tp</i>)	12 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	79 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *feedback query* “**Klasifikasi Keberadaan kalimat**” adalah:

$$Precision \rightarrow P = tp / (tp + fp) = 9 / (9+12) = 9/21 = 0.428$$

$$Recall \rightarrow R = tp / (tp + fn) = 9 / (9+0) = 9/9 = 1$$

4. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Tidak Relevan.

Dari proses *feedback* yang dilakukan maka, berdasarkan Tabel 5.15, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 4 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 13 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak

relevan sebanyak (*tn*) 83 dokumen. Tujuan pengujian ini untuk menilai *precision* dan *recall non-interpolasi* dari hasil pencarian proses *relevance feedback* dengan algoritma genetika terhadap *query* panjang dan dokumen yang dijadikan *feedback* adalah dokumen yang tidak relevan. Tabel 5.17 berikut adalah hasil pengujiannya:

Tabel 5.17. Hasil pengujian *Feedback Presicion* dan *Recall Non-interpolasi* pada Q₄

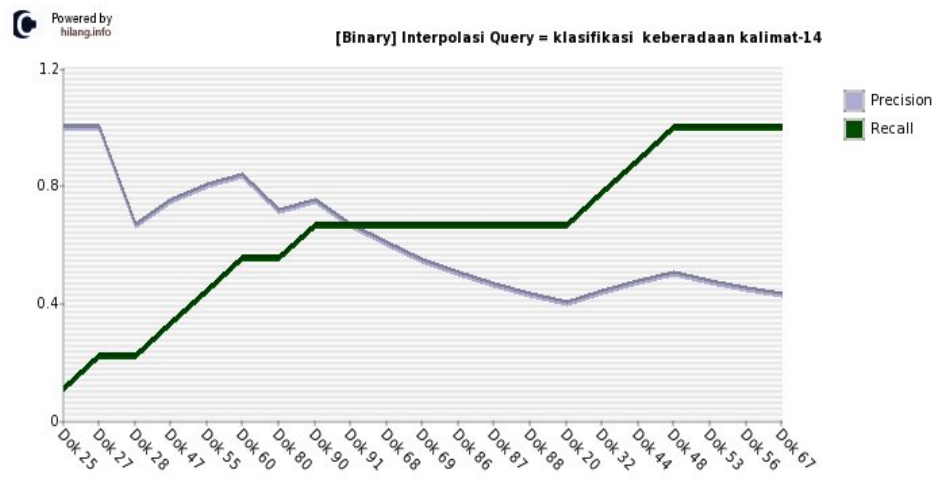
	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	4 (<i>tp</i>)	13(<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	83 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *feedback query* “Klasifikasi Keberadaan kalimat” adalah:

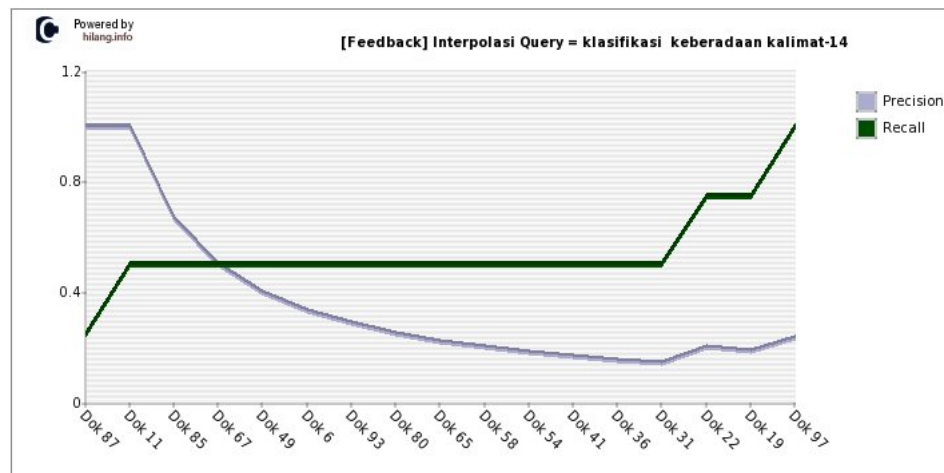
$$Precision \rightarrow P = tp / (tp + fp) = 4 / (4+13) = 4/17 = 0.235$$

$$Recall \rightarrow R = tp / (tp + fn) = 4 / (4+0) = 4/4 = 1$$

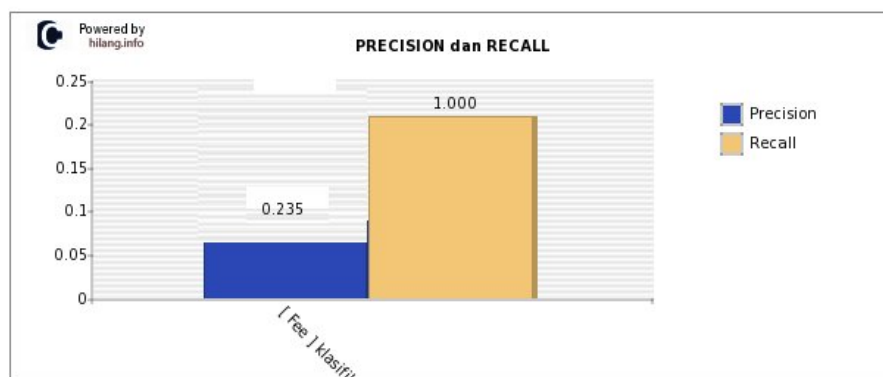
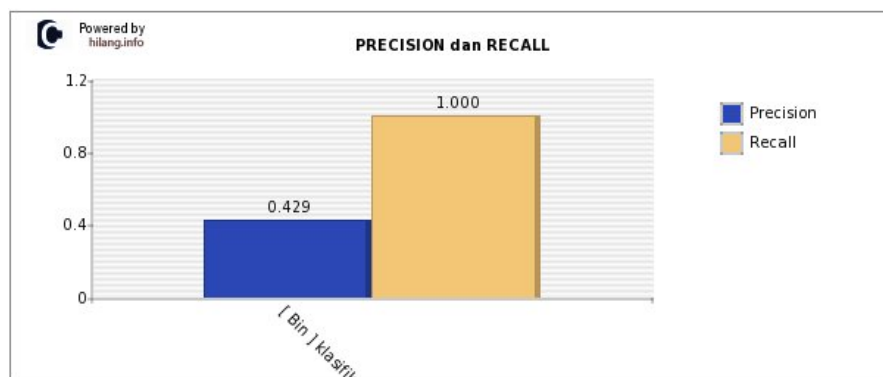
Berikut adalah grafik *precision* dan *recall* interpolasi *query 4*



Gambar 5.23: Grafik *Precision* dan *Recall Interpolasi* untuk *Query* “Klasifikasi Keberadaan Kalimat”



Gambar 5.24: Grafik *Feedback Precision* dan *Recall Interpolasi* untuk *Query* "Klasifikasi Keberadaan Kalimat"



Gambar 5.25: Grafik *Precision* dan *Recall Non-Interpolasi* untuk *Query* "Klasifikasi Keberadaan Kalimat"

1. Hasil Pengujian *Precision* dan *Recall Interpolasi* Terhadap Kata Yang Panjang Untuk *Query 5: Pendeteksi Penyakit*

Pengujian dengan menginputkan kata yang panjang sebagai *query*. Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (dua suku kata). Tabel 5.18 adalah hasil pengujian *precision* dan *recall interpolasi*:

Tabel 5.18 Hasil Pengujian *Precision* dan *Recall Interpolasi query 5*

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	12	Ya	$P = 1/1 = 1$	$R = 1/5 = 0.2$
2	25	Tidak	$P = 1/2 = 1$	$R = 1/5 = 0.2$
3	66	Ya	$P = 2/3 = 1$	$R = 2/5 = 0.4$
4	67	Ya	$P = 3/4 = 1$	$R = 3/5 = 0.6$
5	8	Tidak	$P = 3/5 = 0.8$	$R = 3/5 = 0.6$
6	81	Tidak	$P = 3/6 = 0.833$	$R = 3/5 = 0.6$
7	11	Tidak	$P = 3/7 = 0.857$	$R = 3/5 = 0.6$
8	86	Tidak	$P = 3/8 = 0.875$	$R = 3/5 = 0.6$
9	13	Ya	$P = 4/9 = 0.888$	$R = 4/5 = 0.8$
10	87	Ya	$P = 5/10 = 0.9$	$R = 5/5 = 1$
11	14	Tidak	$P = 5/11 = 0.909$	$R = 5/5 = 1$
12	88	Tidak	$P = 5/12 = 0.916$	$R = 5/5 = 1$
13	15	Tidak	$P = 5/13 = 0.923$	$R = 5/5 = 1$
14	93	Tidak	$P = 5/14 = 0.928$	$R = 5/5 = 1$
15	24	Tidak	$P = 5/15 = 0.866$	$R = 5/5 = 1$
16	27	Tidak	$P = 5/16 = 0.875$	$R = 5/5 = 1$
17	28	Tidak	$P = 5/17 = 0.882$	$R = 5/5 = 1$
18	29	Tidak	$P = 5/18 = 0.888$	$R = 5/5 = 1$
19	42	Tidak	$P = 5/19 = 0.894$	$R = 5/5 = 1$
20	48	Tidak	$P = 5/20 = 0.9$	$R = 5/5 = 1$
21	53	Tidak	$P = 5/21 = 0.904$	$R = 5/5 = 1$
22	55	Tidak	$P = 5/22 = 0.888$	$R = 5/5 = 1$
23	65	Tidak	$P = 5/23 = 0.894$	$R = 5/5 = 1$
24	71	Tidak	$P = 5/24 = 0.9$	$R = 5/5 = 1$
25	73	Tidak	$P = 5/25 = 0.904$	$R = 5/5 = 1$

2. Hasil Pengujian *Precision* dan *Recall* Interpolasi Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Tidak Relevan

Dokumen yang dipilih untuk dilakukan *feedback* adalah dokumen yang tidak relevan yaitu dokumen nomor 25 dengan judul dokumen: **“Desain Konseptual Penggunaan Hiperlink Sebgi Alat Bantu Untuk Informasi Temu Kembali Diperpustakaan”**. Tujuannya adalah menilai *precision* dan *recall* interpolasi dari *relevance feedback* dengan algoritma genetika jika dokumen yang dijadikan *feedback* adalah dokumen yang tidak relevan. Hasil pengujian *relevance feedback* dari dokumen yang tidak relevan yang dijadikan *feedback* seperti pada Tabel 5.19 berikut:

Tabel 5.19: Hasil Pengujian *Interpolasi Feedback* untuk No. Dokumen 25

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	100	Tidak	$P = 0/1 = 1$	$R = 0/7 = 0$
2	87	Tidak	$P = 0/2 = 1$	$R = 0/7 = 0$
3	65	Tidak	$P = 0/3 = 1$	$R = 0/7 = 0$
4	3	Ya	$P = 1/4 = 1$	$R = 1/7 = 0.142$
5	6	Tidak	$P = 1/5 = 0.8$	$R = 1/7 = 0.142$
6	27	Ya	$P = 2/6 = 0.833$	$R = 2/7 = 0.287$
7	95	Tidak	$P = 2/7 = 0.857$	$R = 2/7 = 0.287$
8	84	Ya	$P = 3/8 = 0.875$	$R = 3/7 = 0.428$
9	53	Tidak	$P = 3/9 = 0.888$	$R = 3/7 = 0.428$
10	57	Tidak	$P = 3/10 = 0.9$	$R = 3/7 = 0.428$
11	16	Tidak	$P = 2/11 = 0.909$	$R = 3/7 = 0.428$
12	49	Tidak	$P = 3/12 = 0.916$	$R = 3/7 = 0.428$
13	45	Ya	$P = 4/13 = 0.923$	$R = 4/7 = 0.5$
14	34	Tidak	$P = 4/14 = 0.928$	$R = 4/7 = 0.5$
15	69	Tidak	$P = 4/15 = 0.866$	$R = 4/7 = 0.5$
16	74	Tidak	$P = 4/16 = 0.875$	$R = 4/7 = 0.5$
17	77	Ya	$P = 5/17 = 0.882$	$R = 5/7 = 0.714$
18	38	Ya	$P = 6/18 = 0.888$	$R = 6/7 = 0.857$
19	12	Ya	$P = 7/19 = 0.894$	$R = 7/7 = 1$

3. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Terhadap Kata Yang Pendek Untuk *Query 5*

Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall non-interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (dua suku kata). Maka berdasarkan Tabel 5.18, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query (tp)* sebanyak 5 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 20 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query (fn)* sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 75 dokumen. Tabel 5.20 berikut adalah hasil pengujiannya:

Tabel 5.20: Hasil Pengujian *Precision* dan *Recall Non-interpolasi* pada Q5

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	5 (<i>tp</i>)	20 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	75 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *feedback query* “Pendeteksi Penyakit” adalah:

$$Precision \rightarrow P = tp / (tp + fp) = 5 / (5+20) = 5/25 = 0.2$$

$$Recall \rightarrow R = tp / (tp + fn) = 5 / (5+0) = 5/5 = 1$$

4. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Dari *Relevance Feedback* Terhadap *Query Panjang* dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Tidak Relevan

Dari proses *feedback* yang dilakukan maka, berdasarkan Tabel 5.19, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query (tp)* sebanyak 7 dokumen, sedangkan dokumen yang tidak relevan (*fp*)

sebanyak 12 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 81 dokumen. Tujuan pengujian ini untuk menilai *precision* dan *recall non-interpolasi* dari hasil pencarian proses *relevance feedback* dengan algoritma genetika terhadap *query* panjang dan dokumen yang dijadikan *feedback* adalah dokumen yang tidak relevan. Tabel 5.21 berikut adalah hasil pengujiannya:

Tabel 5.21: Hasil pengujian *Feedback Precision* dan *Recall Non-interpolasi* pada Q_5

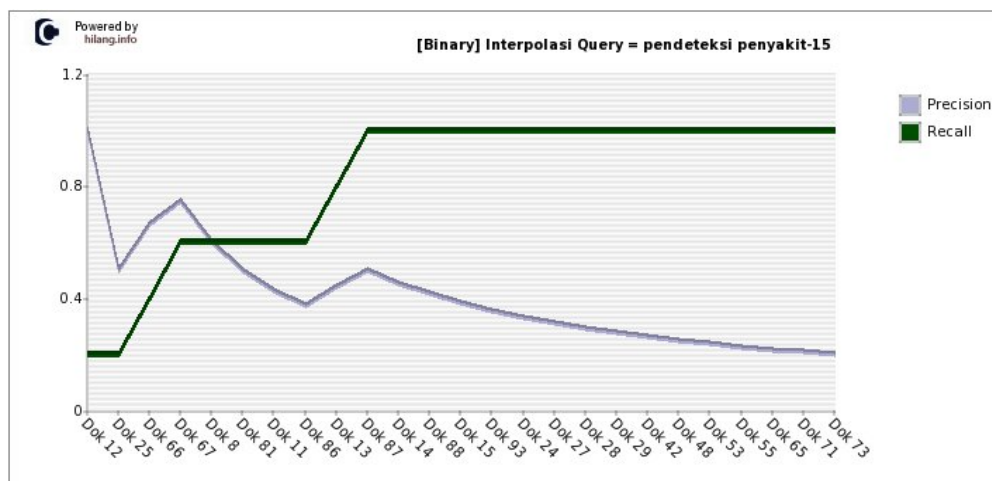
	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	7 (<i>tp</i>)	12 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	81 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *feedback query* “Pendeteksi Penyakit” adalah:

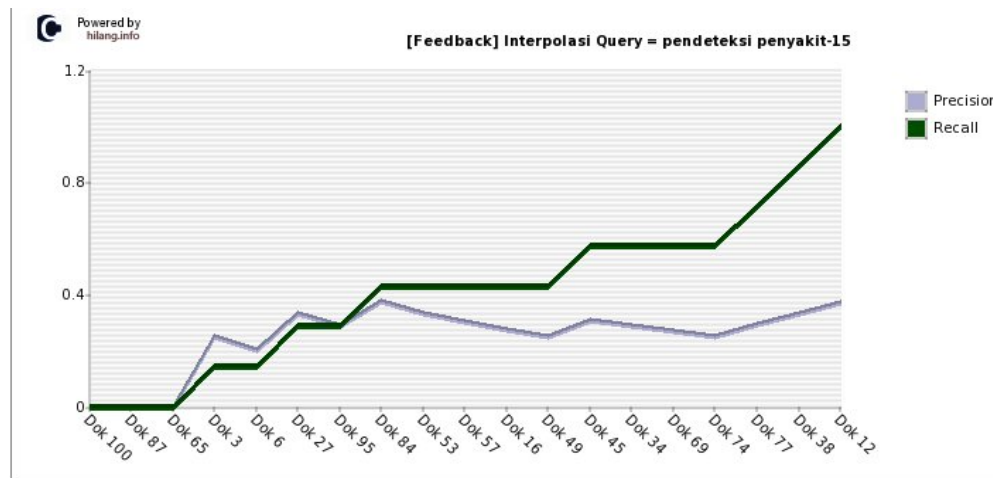
$$\text{Precision} \rightarrow P = tp / (tp + fp) = 7 / (7+12) = 7/19 = 0.368$$

$$\text{Recall} \rightarrow R = tp / (tp + fn) = 7 / (7+0) = 7/7 = 1$$

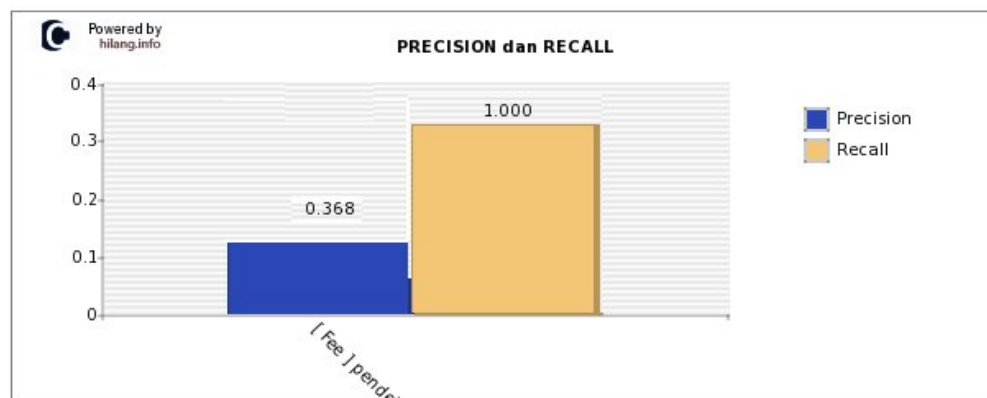
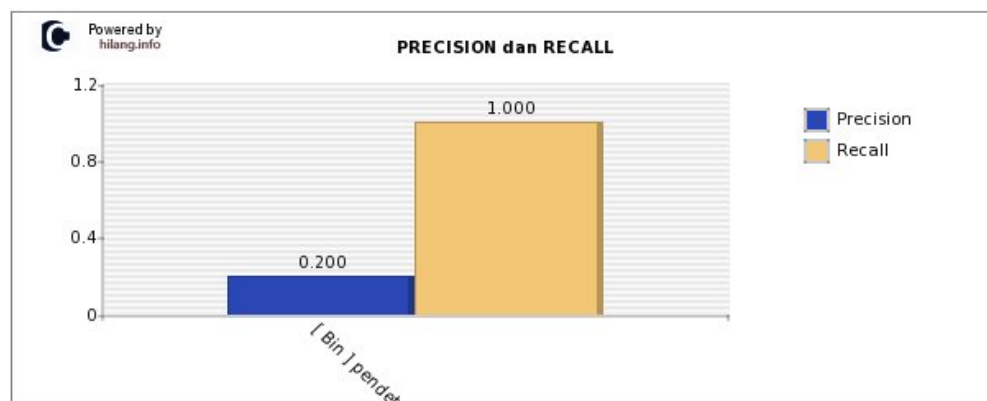
Berikut adalah grafik *precision* dan *recall interpolasi query 5*:



Gambar 5.26: Grafik *Precision* dan *Recall Interpolasi* untuk *Query* “Pendeteksi Penyakit”



Gambar 5.27: Grafik *Feedback Precision* dan *Recall Interpolasi* untuk *Query* "Pendeteksi Penyakit"



Gambar 5.28: Grafik *Precision* dan *Recall Non-Interpolasi* untuk *Query* "Pendeteksi Penyakit"

1. Hasil Pengujian *Precision* dan *Recall Interpolasi* Terhadap Kata Yang Panjang Untuk *Query* 6: Jaringan Saraf Tiruan

Pengujian dengan menginputkan kata yang panjang sebagai *query* dilakukan dengan menginputkan *query* yang tiga suku kata. Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (tiga suku kata). Tabel 5.22 berikut adalah hasil pengujian untuk *query* “Jaringan Saraf Tiruan”:

Tabel 5.22: Hasil Pengujian *Precision* dan *Recall Interpolasi* *query* 6

No	No Dokumen	Relevan?	<i>Precision</i> (P)	<i>Recall</i> (R).
1	12	Tidak	$P = 0/1 = 0$	$R = 0/4 = 0$
2	37	Ya	$P = 1/2 = 0.5$	$R = 1/4 = 0.25$
3	44	Ya	$P = 2/3 = 0.666$	$R = 2/4 = 0.5$
4	77	Ya	$P = 3/4 = 0.75$	$R = 3/4 = 0.75$
5	87	Ya	$P = 4/5 = 0.8$	$R = 4/4 = 1$
6	11	Tidak	$P = 4/6 = 0.666$	$R = 4/4 = 1$
7	13	Tidak	$P = 4/7 = 0.571$	$R = 4/4 = 1$
8	20	Tidak	$P = 4/8 = 0.5$	$R = 4/4 = 1$
9	22	Tidak	$P = 4/9 = 0.444$	$R = 4/4 = 1$
10	55	Tidak	$P = 4/10 = 0.4$	$R = 4/4 = 1$
11	57	Tidak	$P = 4/11 = 0.363$	$R = 4/4 = 1$
12	71	Tidak	$P = 4/12 = 0.333$	$R = 4/4 = 1$
13	72	Tidak	$P = 4/13 = 0.307$	$R = 4/4 = 1$

2. Hasil Pengujian *Precision* dan *Recall Interpolasi* Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Relevan.

Dokumen yang dipilih untuk dilakukan *feedback* adalah dokumen yang relevan yaitu nomor dokumen 37 dengan judul dokumen: “Identifikasi Tanda

Tangan Menggunakan Jaringan Syaraf Tiruan Perambatan Balik (Backpropagation)”. Tujuannya adalah menilai *precision* dan *recall* interpolasi dari *relevance feedback* dengan algoritma genetika jika dokumen yang dijadikan *feedback* adalah dokumen yang relevan. Hasil pengujian *relevance feedback* dari dokumen yang relevan yang dijadikan *feedback* dapat dilihat pada Tabel 5.23 sebagai berikut:

Tabel 5.23: Hasil Pengujian *Interpolasi Feedback* untuk No. Dokumen 37

No	No Dokumen	Relevan?	<i>Precision (P)</i>	<i>Recall (R).</i>
1	57	Tidak	$P = 0/1 = 1$	$R = 0/4 = 0$
2	85	Tidak	$P = 0/2 = 1$	$R = 0/4 = 0$
3	5	Ya	$P = 1/3 = 0.666$	$R = 1/4 = 0.25$
4	92	Ya	$P = 2/4 = 0.5$	$R = 2/4 = 0.5$
5	77	Ya	$P = 3/5 = 0.4$	$R = 3/4 = 0.75$
6	74	Tidak	$P = 3/6 = 0.333$	$R = 3/4 = 0.75$
7	67	Tidak	$P = 3/7 = 0.285$	$R = 3/4 = 0.75$
8	44	Ya	$P = 4/8 = 0.25$	$R = 4/4 = 1$
9	11	Tidak	$P = 4/9 = 0.222$	$R = 4/4 = 1$
10	101	Tidak	$P = 4/10 = 0.3$	$R = 4/4 = 1$

3. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Terhadap Kata Yang Pendek Untuk *Query* 6

Tujuan dari pengujian ini adalah mengukur kemampuan model *binary independence* dengan menilai *precision* dan *recall non-interpolasi* dari hasil pencarian yang menginputkan *query* yang panjang (tiga suku kata). Maka berdasarkan Tabel 5.22, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 4 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 9 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 83 dokumen. Terlihat pada Tabel 5.24:

Tabel 5.24: Hasil Pengujian *Precision* dan *Recall Non-interpolasi* pada Q6

	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	4 (<i>tp</i>)	9 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	83 (<i>tn</i>)

Maka, nilai *precision* dan *recall* untuk *query* “Jaringan Saraf Tiruan” adalah:

$$Precision \rightarrow P = tp / (tp + fp) = 4 / (4+9) = 4/13 = 0.307$$

$$Recall \rightarrow R = tp / (tp + fn) = 4 / (4+0) = 4/4 = 1$$

4. Hasil Pengujian *Precision* dan *Recall Non-interpolasi* Dari *Relevance Feedback* Terhadap *Query* Panjang dan Dokumen Yang Dijadikan *Feedback* Adalah Dokumen Yang Relevan

Dari proses *feedback* yang dilakukan maka, berdasarkan Tabel 5.23, ditunjukkan bahwa jumlah dokumen yang dikembalikan yang relevan dengan *query* (*tp*) sebanyak 4 dokumen, sedangkan dokumen yang tidak relevan (*fp*) sebanyak 6 dokumen. Dan untuk jumlah dokumen yang tidak dikembalikan yang relevan dengan *query* (*fn*) sebanyak 0 dokumen, sedangkan dokumen yang tidak relevan sebanyak (*tn*) 90 dokumen. Tujuan pengujian ini untuk menilai *precision* dan *recall non-interpolasi* dari hasil pencarian proses *relevance feedback* dengan algoritma genetika terhadap *query* panjang dan dokumen yang dijadikan *feedback* adalah dokumen yang relevan. Tabel 5.25 berikut adalah hasil pengujiannya:

Tabel 5.25: Pengujian *Feedback Precision* dan *Recall Non-interpolasi* pada Q6

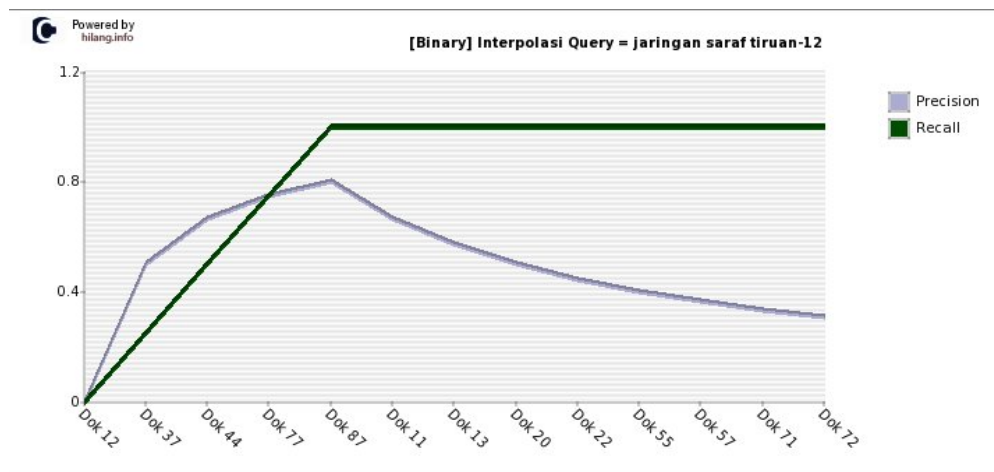
	<i>Relevant</i>	<i>Non-relevant</i>
<i>Retrieved</i>	4 (<i>tp</i>)	9 (<i>fp</i>)
<i>Not retrieved</i>	0 (<i>fn</i>)	83 (<i>tn</i>)

Maka, nilai precision dan recall untuk *feedback* query “Jaringan Saraf Tiruan” adalah:

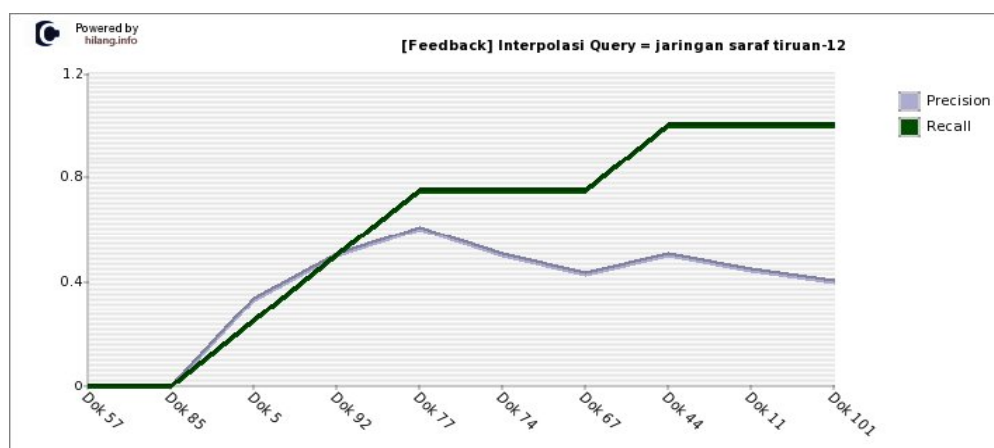
$$\text{Precision} \rightarrow P = tp / (tp + fp) = 4 / (4+6) = 4/10 = 0.4$$

$$\text{Recall} \rightarrow R = tp / (tp + fn) = 4 / (4+0) = 4/4 = 1$$

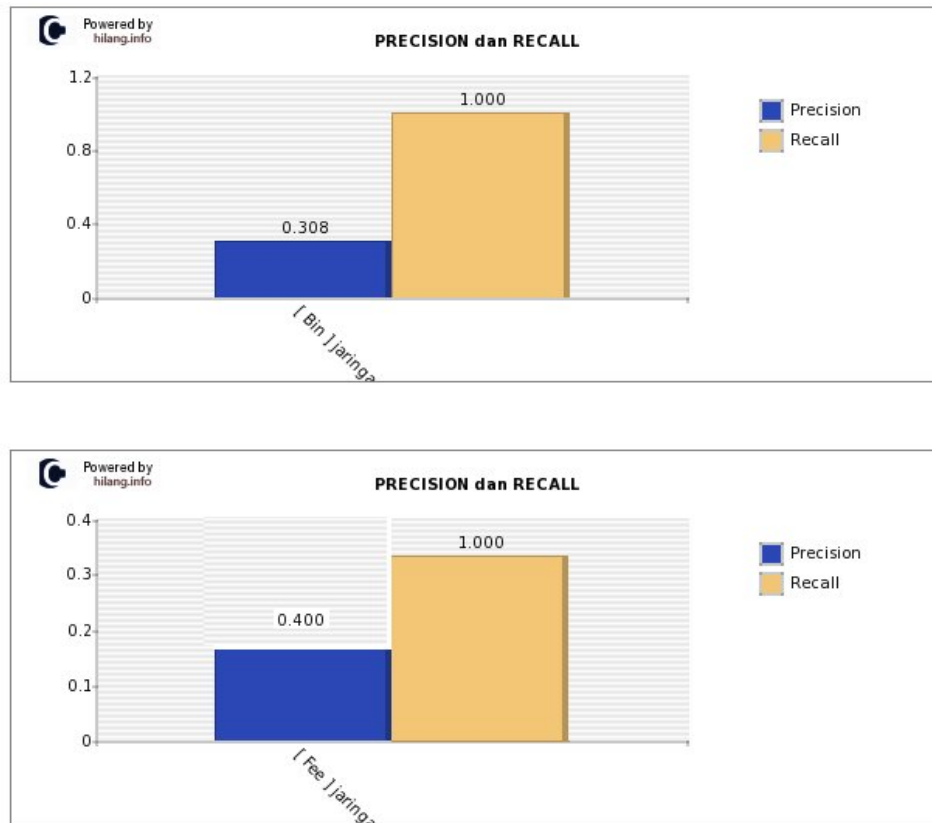
Berikut adalah grafik *precision* dan *recall* interpolasi query 6:



Gambar 5.29 : Grafik *Precision* dan *Recall* Interpolasi untuk Query “Jaringan Saraf Tiruan”



Gambar 5.30: Grafik *Feedback Precision* dan *Recall* Interpolasi untuk Query “Jaringan Saraf Tiruan”



Gambar 5.31: Grafik *Precision* dan *Recall* Non-Interpolasi untuk *Query* "Jaringan Saraf Tiruan"

5.2.4 Kesimpulan Pengujian Unjuk Kerja Sistem

Pada Tabel 5.26 berikut adalah tabel hasil pengujian *precision* dan *recall* penerapan *relevance feedback* dalam sistem temu kembali informasi dengan *binary independence model* berdasarkan panjang pendek *query* dan relevan tidak relevan dokumen yang dijadikan *feedback*:

Tabel 5.26: Hasil pengujian *Precision* dan *Recall*

No	Query Pengujian	Status Query	Model Binary Independence		Status Dokumen Feedback	Relevance Feedback	
			Precision	Recall		Precision	Recall
1	Genetika	Query Pendek	100%	100%	Dokumen Relevan	100%	100%
2	Enkripsi	Query Pendek	83.3%	100%	Dokumen Tidak Relevan	76.4%	100%
3	Kecerdasan Buatan	Query Panjang	63.6%	100%	Dokumen Relevan	35%	100%
4	Klasifikasi Keberadaan Kalimat	Query Panjang	42.8%	100%	Dokumen Tidak Relevan	23.5%	100%
5	Pendeteksi Penyakit	Query Panjang	20%	100%	Dokumen Tidak Relevan	36.8%	100%
6	Jaringan Saraf Tiruan	Query Panjang	30.7%	100%	Dokumen Tidak Relevan	40%	100%

Hasil rata-rata pengujian yang *precision* dan *recall* yang diperoleh dari penerapan *relevance feedback* pada sistem temu kembali informasi yang menggunakan *binary independence model* seperti pada perhitungan sebagai berikut:

1. Dari seluruh pengujian maka rata-rata nilai *precision* dari sistem temu kembali informasi dengan model *binary independence* adalah:

$$\text{Rata-rata } precision = (\text{precision Pengujian 1} + \text{precision Pengujian 2} + \text{precision Pengujian 3} + \text{precision Pengujian 4} + \text{precision Pengujian 5} + \text{precision Pengujian 6}) / 6 \times 100\%$$

$$= (P_{Q1} + P_{Q2} + P_{Q3} + P_{Q4} + P_{Q5} + P_{Q6}) / 6 \times 100\%$$

$$= (1+0.833+0.724+0.307+0.428+0.2):6 \times 100\%$$

$$= 3.692:6 \times 100\% = 61.533\%$$

$$\text{Rata-rata } recall = (recall \text{ Pengujian 1} + recall \text{ Pengujian 2} + recall \text{ Pengujian 3} + recall \text{ Pengujian 4} + recall \text{ Pengujian 5} + recall \text{ Pengujian 6}) / 6 \times 100\%$$

$$= (R_{Q1} + R_{Q2} + R_{Q3} + R_{Q4} + R_{Q5} + R_{Q6}) / 6 \times 100\%$$

$$= (1+1+1+1+1+1):6 \times 100\% = 100\%$$

2. Dari seluruh pengujian maka rata-rata nilai *precision* dari sistem temu kembali informasi dengan *relevance feedback* dengan algoritma genetika adalah:

$$\text{Rata-rata } precision = (precision \text{ Pengujian 1} + precision \text{ Pengujian 2} + precision \text{ Pengujian 3} + precision \text{ Pengujian 4} + precision \text{ Pengujian 5} + precision \text{ Pengujian 6}) / 6 \times 100\%$$

$$= (P_{Q1} + P_{Q2} + P_{Q3} + P_{Q4} + P_{Q5} + P_{Q6}) : 6 \times 100\%$$

$$= (1+0.764+0.35+0.4+0.235+0.368):6 \times 100\%$$

$$= 3.117:6 \times 100\% = 51.95\%$$

$$\text{Rata-rata } recall = (recall \text{ Pengujian 1} + recall \text{ Pengujian 2} + recall \text{ Pengujian 3} + recall \text{ Pengujian 4} + recall \text{ Pengujian 5} + recall \text{ Pengujian 6}) / 6 \times 100\%$$

$$= (R_{Q1} + R_{Q2} + R_{Q3} + R_{Q4} + R_{Q5} + R_{Q6}) : 6 \times 100\%$$

$$= (1+1+1+1+1+1):6 \times 100\% = 100\%$$

BAB VI

PENUTUP

Pada bab ini akan diuraikan beberapa kesimpulan dari hasil yang didapatkan selama penelitian dan saran yang dapat digunakan pada penelitian selanjutnya.

6.1 Kesimpulan

Setelah menyelesaikan tahapan-tahapan penelitian sistem temu kembali informasi dan proses *relevan feedback* dapat diambil beberapa kesimpulan, yaitu :

1. Berdasarkan penelitian yang telah dilakukan, model *binary independence* yang digunakan memberikan hasil yaitu dengan rata-rata *precision* 61.53% dan *Recall* 100% dari hasil 6 kali percobaan. Hasil pencarian *Binary independence model* dipegaruhi oleh sifat *binary independence model* yaitu sifat dari *binary* dimana pembobotan mengabaikan jumlah kemunculan *term*, hanya mempertimbangkan muncul atau tidak dan sifat dari *independence* yaitu mengansumsikan *term* tidak saling berkaitan.
2. Metode *relevance feedback* dengan algoritma genetika yang diterapkan dalam sistem temu kembali informasi ini mampu memberikan hasil pencarian dokumen-dokumen yang sama dengan dokumen yang dijadikan *feedback*. Dengan hasil pengujian *precision* dan *recall* mengalami penurunan yaitu rata-rata *precision* 51.95% dan rata-rata *recall* 100% dari hasil 6 kali percobaan, hal ini lebih disebabkan karena dokumen yang dijadikan *feedback* belum tentu dokumen yang relevan dan banyaknya *term* yang sama dengan dokumen yang dijadikan *feedback* belum tentu mewakili dari *query* yang dimasukan pengguna. Metode *relevance feedback* akan menurunkan nilai *precision* dan *recall* apabila pencarian awal menghasilkan memiliki nilai kerelevanan yang rendah.

3. Panjang *query* yang dimasukkan untuk pencarian mempengaruhi kualitas korelevanan hasil pencarian. Semakin panjang *query* yang dimasukkan akan menghasilkan semakin banyak pula dokumen yang berhasil ditemukan, dimana dokumen tersebut belum tentu mewakili tujuan dari *query* yang dimasukkan.

6.2 Saran

Berdasarkan penelitian yang telah dilakukan, adapun saran-saran yang dapat dilakukan untuk perbaikan dan pengembangan *relevance feedback* dalam sistem temu kembali informasi mendatang, yaitu :

1. Sistem temu kembali informasi yang dibangun sebaiknya menggunakan model-model lain seperti model OKPBM25, Model ruang *vector* dan *Bayesian Network Model*.
2. Metode *relevance feedback* dapat dikombinasikan dengan metode perluasan *query* dengan menggunakan *Thesaurus* yang menyediakan informasi berdasarkan sinonim dan kata-kata yang saling berhubungan serta frase-frase, sehingga dapat memberikan hasil pencarian dokumen yang lebih banyak dan lebih relevan dan metode *relevance feedback* dapat menggunakan algoritma lain atau metode lain, seperti *cosine*, *dice*, dan *jaccard*.

DAFTAR PUSTAKA

- Adisantoso, J., Ridha A., *Corpus Dokumen Teks Bahasa Indonesia Untuk Pengujian Efektifitas Temu Kembali Informasi*, Laporan Hibah Penelitian SP4, Departemen Ilmu Komputer FMIPA IPB, Bogor, 2004.
- Adisantoso, J., Ridha A., Agusetyawan, A. W., *Relevance Feedback Pada Temu-Kembali Teks Berbahasa Indonesia Dengan Metode IDE-DEC-HI Dan IDE-REGULAR*, Bogor, 2004.
- Anhar, A. S., *Information Retrieval System Dengan Menggunakan Algoritma Genetika Pada Tugas Akhir*, Pekanbaru, 2012
- Basuki, A. *Algoritma Genetika Suatu Alternatif Penyelesaian Permasalahan*, 2003.
- Cios, Krzstof J. Etc., *Data Mining A Knowledge Discovery Approach*, Springer, 2007.
- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze, *Introduction to Information Retrieval*, Cambridge University Press, 2008.
- Jaya, H. “*Perbandingan Performansi Word Indexing dan Phrase Indexing dalam Sistem Temu Balik Informasi dengan Menggunakan Model Probabilistik.*” Skripsi Terpublikasi. Bandung: Institut Teknologi Bandung, 2007.
- Kusumadewi, S. *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu, 2003.
- Mandala, R. *Evaluasi Efektifitas Metode Machine-Learning pada Search-Engine*, Seminar Nasional Aplikasi Teknologi Informasi, ISSN: 1907-5022. Yogyakarta, 2006.
- Mandala, R., Hendra, S., *Peningkatan Peformasi Sistem Temu Kembali Informasi dengan Perluasan Query Secara Otomatis*, Laboratorium Keahlian Informatika Teori Departement Teknik Informatika, Institut Teknologi Bandung, 2006.
- Ramadhany, T. *Implementasi Kombinasi Model Ruang Vektor dan Model Probabilistik pada Sistem Temu Kembali Informasi*. Teknologi Bandung, 2008.

Sinclair, J. *A Guide to Good Practice Corpus and Text-Basic Principles*. Tuscan Word Centre, 2004.

Trunojoyo, H., *Sistem Temu Balik Informasi* (sebuah contoh implementasi), 2010.

Wakid, dkk. *Sistem Temu Kembali Berbasis Fitur Warna dan Tekstur*, ITS, Surabaya 2011.